RĪGAS TEHNISKĀ UNIVERSITĀTE

Tianhua Chen

INTERAKTĪVO MEDIJU TRAFIKA PRECĪZĀ NOVĒROŠANA UN KLASIFIKĀCIJA

Promocijas darba kopsavilkums

RTU Izdevniecība
Rīga 2026

RĪGAS TEHNISKĀ UNIVERSITĀTE
Datorzinātnes, informācijas tehnoloģijas un enerģētikas fakultāte
Fotonikas, elektronikas un elektronisko sakaru institūts

Tianhua Chen
Doktora studiju programmas "Telekomunikācijas" doktorants

INTERAKTĪVO MEDIJU TĪKLA TRAFIKA PRECĪZĀ NOVĒROŠANA UN KLASIFIKĀCIJA

Promocijas darba kopsavilkums

Zinātniskie vadītāji

asociētais profesors *Dr. sc. ing.*
ELANS GRABS

asociētais profesors *Dr. sc. ing.*
ALEKSANDRS IPATOVS

Līdzkonsultante
profesore *Ph. D.*
CANO BAÑOS, MARÍA DOLORES

RTU Izdevniecība
Rīga 2026

Chen, T. Interaktīvo mediju trafika precīzā novērošana un klasifikācija. Promocijas darba kopsavilkums. Rīga: RTU Izdevniecība, 2026. 52 lpp.

Publicēts saskaņā ar promocijas padomes "RTU P-08" 2025. gada 12.. novembrī lēmumu, protokols Nr. 43.

Promocijas darbs tapis ar Eiropas Savienības Atveseļošanas un noturības mehānisma projekta Nr. 5.2.1.1.i.0/2/24/I/CFLA/003 "Konsolidācijas un pārvaldības izmaiņu īstenošana Rīgas Tehniskajā universitātē, Liepājas Universitātē, Rēzeknes Tehnoloģiju akadēmijā, Latvijas Jūras akadēmijā un Liepājas Jūrniecības koledžā virzībai uz izcilību augstākajā izglītībā, zinātnē un inovācijā" (ID 1072) atbalstu.

Vāka attēls radīts, izmantojot *ChatGPT*.

https://doi.org/10.7250/9789934372537
ISBN 978-9934-37-253-7 (pdf)

PROMOCIJAS DARBS IZVIRZĪTS ZINĀTNES DOKTORA GRĀDA IEGŪŠANAI RĪGAS TEHNISKAJĀ UNIVERSITĀTĒ

Promocijas darbs zinātnes doktora (*Ph. D.*) grāda iegūšanai tiek publiski aizstāvēts 2026. gada 13. februārī plkst. 11.00 Rīgas Tehniskās universitātes Datorzinātnes, informācijas tehnoloģijas un enerģētikas fakultātē, Āzenes ielā 12, 201. auditorijā.

OFICIĀLIE RECENZENTI

Profesors *Dr. sc. ing.* Sandis Spolītis,
Rīgas Tehniskā universitāte

Profesors *Dr. sc. ing. Jaime Lloret,*
Valensijas Politehniskā universitāte, Spānija

Profesors *Dr. sc. ing. Mario Alberto Montagut Climent,*
Valensijas Universitāte, Spānija

APSTIPRINĀJUMS

Apstiprinu, ka esmu izstrādājis šo promocijas darbu, kas iesniegts izskatīšanai Rīgas Tehniskajā universitātē zinātnes doktora (*Ph. D.*) grāda iegūšanai. Promocijas darbs zinātniskā grāda iegūšanai nav iesniegts nevienā citā universitātē.

Tianhua Chen ……. (paraksts)
Datums: ……

Promocijas darbs ir uzrakstīts angļu valodā, tajā ir ievads, četras sadaļas, secinājumi, literatūras saraksts, 16 attēli, 23 tabulas, astoņi pielikumi, kopā 143 lappuses, ieskaitot pielikumus. Literatūras sarakstā ir 157 nosaukumi.

SATURS

IZMANTOTIE SAĪSINĀJUMI

AAC	*Advanced Audio Coding*
ABR	*Adaptive BitRate*
AI	*Artificial Intelligence*
BiRNN	*Bi-directional Recurrent Neural Network*
CBR	*Constant BitRate*
CCDF	*Complementary Cumulative Distribution Function*
CDN	*Content Delivery Network*
CNN	*Convolutional Neural Network*
CV	*Coefficient of Variation*
DASH	*Dynamic Adaptive Streaming over HTTP*
Db	*Daubechies*
DBN	*Deep Belief Network*
DLSS	*Deep Learning Super Sampling*
DPI	*Deep Packet Inspection*
DTLS	*Datagram Transport Layer Security*
EBSNN	*Extended Byte Segment Neural Network*
ENN	*Edited Nearest Neighbors*
FPS	*Frames Per Second*
GCN	*Graph Convolutional Network*
GI	*General Independent*
GNN	*Graph Neural Network*
GOP	*Group of Pictures*
GRU	*Gated Recurrent Unit*
GSO	*Generic Segmentation Offload*
GTP-U	*GPRS Tunnelling Protocol User Plane*
HLS	*HTTP Live Streaming*
HTTP	*Hyper Text Transfer Protocol*
HTTP-FLV	*HTTP FLV live stream*
IAT	*InterArrival Time*
ICE	*Interactive Connectivity Establishment*
IDC	*International Data Corporation*
IEC	*International Electrotechnical Commission*
IP	*Internet Protocol*
ISO	*International Organization for Standardization*
ISP	*Internet Service Providers*
KNN	*K Nearest Neighbors*
LAN	*Local Area Network*
LSTM	*Long Short Term Memory*
MAC	*Media Access Control*
MLP	*Multi-Layer Perceptron*
MPD	*Media Presentation Description*
MSE	*Mean Squared Error*
MTU	*Maximum Transmission Unit*
NAT	*Network Address Translation*
NGAP	*Next Generation Application Protocol*
NTC	*Network Traffic Classification*
NTMA	*Network Traffic Monitoring and Analysis*
OTT	*Over The Top*
PCAP	*Packet Capture*
PDF	*Probability Density Function*
QoE	*Quality of Experience*
QoS	*Quality of Service*
QUIC	*Quick UDP Internet Protocol*
RAN	*Radio Access Network*
RFC	*Request for Comments*
RLC	*Radio Link Control*
RNN	*Recurrent Neural Network*
RTCP	*Real Time Control Protocol*
RTMP	*Real Time Messaging Protocol*
RTP	*Real-Time Protocol*
SMOTE	*Synthetic Minority Oversampling TEchnique*
SNI	*Server Name Indication*
SRS	*Simple Realtime Server*
SRT	*Secure Reliable Transport*
SSIM	*Structural Similarity Index*
SSL	*Secure Sockets Layer*
STUN	*Session Traversal Utilities for NAT*
TCP	*Transmission Control Protocol*
TFSN	*TLS Flow Sequence Network*
TLS	*Transport Layer Security*
Tor	*The onion router*
TPR	*True Positive Rate*
TSO	*TCP Segmentation Offload*
TURN	*Traversal Using Relays around NAT*
UHD	*Ultra High Definition*
UPF	*User Plane Function*
VBR	*Variable BitRate*
VDS	*Virtual Desktop Streaming*
VLC	*VideoLAN Client*
VLS	*Video Live Streaming*
VoD	*Video on Demand*
VPN	*Virtual Private Network*
WebRTC	*Web Real Time Communication*
XR	*eXtended Reality*

TABULU SARAKSTS

1. PĀRSKATS

1.1. Ievads

Saskaņā ar Starptautiskās datu korporācijas (*International Data Corporation – IDC*) prognozēm līdz 2025. gada beigām interneta lietotāji visā pasaulē datu plūsmās saģenerēs 175 zettabaitus (ZB) [1]. Tīkla datu plūsma ir sarežģīts galapunktu un to mijiedarbību kopums, ko raksturo sevlīdzīgums (*self-similarity*), sprādzienveida (*bursty*) raksturs un cikliskās likumsakarības. Strauji pieaugot datu apjomam, tīkla datu plūsmas uzraudzībai un analīzei (*Network Traffic Monitoring and Analysis – NTMA*) ir būtiska nozīme tīkla veiktspējas kontrolē un resursu izmantošanas uzlabošanā [2], [3]. Viena no svarīgākajām *NTMA* sastāvdaļām ir tīkla datu plūsmas klasifikācija (*Network Traffic Classification – NTC*). *NTC* ir process, kurā tīkla datu plūsmas tiek kategorizētas, balstoties to īpašībās, lai identificētu dažādas lietotnes vai pakalpojumus, kas darbojas tīklā [4]–[12]. Šī klasifikācija ir būtiska interneta pakalpojumu sniedzējiem (*Internet Service Providers – ISP*), lai efektīvi pārvaldītu savu infrastruktūru. Kategorizējot datu plūsmu pēc prioritātes līmeņiem, tiek nodrošināts, ka kritiskās lietotnes saņem pietiekamu joslas platumu un skaitļošanas resursus. Turklāt šī pieeja veicina anomāliju atklāšanu un tīkla drošības un veiktspējas uzturēšanas pasākumu ieviešanu. Video pakalpojumi kļuvuši par galveno datu plūsmas veidotāju [13]. Saskaņā ar *Cisco* datiem līdz 2021. gadam video saturs veidoja 82 % no visa *IP* datu plūsmas apjoma [14]. *OTT* (*Over the top*) lietotnes, kas darbojās virs *IP* tīkla, piemēram, *Netflix* un *YouTube*, dominē šajā pieaugumā – *Sandvine Incorporated* ziņoja par 24% video datu plūsmas pieaugumu 2023. gada janvārī [15], [16]. Mūsdienās daudzas populārākās *OTT* platformas piedāvā 4 K/8 K ultraaugstās izšķirtspējas (*Ultra High Definition – UHD*) saturu ar augstu kadru skaitu sekundē, piemēram, 60 *FPS* vai 120 *FPS*, kas būtiski ietekmē tīkla datu plūsmas apjomu [17]–[24].

Lielākā daļa esošo pētījumu par mediju lietotņu datu plūsmas uzraudzību un klasifikāciju galvenokārt fokusējās uz šifrētām *OTT* video lietotnēm [25]–[28]. Pētnieki ir apkopojuši lielu tīkla datu plūsmas apjomu un izveidojuši vairākas publiski pieejamas datu kopas (*ISCXVPN2016, ISCXTor2016, MIRAGE-2019, CICDarknet2020, MAppGraph2021* un *UTMobileNet2021*) [29]–[34]. Šīs datu kopas ir marķētas, pamatojoties uz dažādām pazīmēm, piemēram, nosaukumu, straumēšanas platformu, audio un video saturu, iekapsulēšanas (*encapsulation*) metodi, šifrēšanu, anonimizācijas metodēm, pārsūtīšanas protokoliem, pakalpojuma kvalitātes līmeņiem (*Quality of Service – QoS*), lietotāja pieredzes kvalitāti (*Quality of Experience – QoE*) un citām īpašībām [35]–[41]. Izmantojot modernus mašīnmācīšanās un dziļās mācīšanās klasifikatorus, šajos pētījumos tika sasniegta izcila klasifikācijas precizitāte dažādos detalizācijas līmeņos, tostarp analizējot datu vērtumu (*payload*) un datu plūsmas (*flows*) [42]–[86]. Lai gan šajās datu kopās ir iekļauti ar video saistīti dati, tie parasti veido tikai nelielu daļu no kopējā datu apjoma. Dažās patentētās, publiski nepieejamās datu kopās galvenais akcents veikts uz video datu plūsmas, atlasot tās ar klasifikācijas marķējumu palīdzību, kas iegūti no metadatiem, piemēram, atskaņošanas paradumiem un video parametriem [87]–[94]. Tomēr šādam datu kopām joprojām ir vairāki ierobežojumi. Bieži vien pastāv vāja korelācija un nepietiekami izskaidrojama saistība starp

nolasēm un to patiesajām (*ground-truth*) vērtībām, klasifikācijas robežas ir neskaidras, dati bieži vien ir sajaukti ar citu lietotņu datu plūsmām. Šī pārklāšanās notiek vairākās dimensijās, tostarp tīkla protokolos, sistēmu arhitektūrās, lietotņu kategorijās un datu plūsmas izplatīšanas modeļos, kas padara mediju lietotņu datu plūsmas profilus vēl sarežģītākus [95], [96].

Strauji attīstās jaunas lietotnes, piemēram, mākoņspēles, sociālie tīkli, tiešsaistes semināri un paplašinātā realitāte (*Extended Reality – XR*/*Metaverse*) [97]–[99]. Lielākā daļa esošo pētījumu fokusējas uz datu plūsmas modelēšanas un analīzes, savukārt uzraudzība un klasifikācija jaunajām lietotnēm joprojām atrodas agrīnā izpētes stadijā [100]–[108]. Tomēr šīs jaunās lietotnes attīstās tik strauji, ka kļūst arvien grūtāk kategorizēt tīkla datu plūsmas, izmantojot tradicionālās metodes, kas balstītas lietotnēs vai pakalpojumos. Lietotņu savstarpējās mijiedarbības un arvien pieaugošās interaktīvās sastāvdaļas ietekmē datu plūsmu īpašības kļūst arvien līdzīgākas, tādējādi samazinot atšķirības starp dažādām lietotņu kategorijām. Lai iegūtu padziļinātu izpratni par nākamās paaudzes lietotņu datu plūsmas raksturu, ir jāveic detalizēta (*fine-grained*) interaktīvo mediju tīkla datu plūsmas uzraudzība un klasifikācija. *OTT* pakalpojumu sniedzējiem tādā veidā ir iespēja prognozēt interaktīvo mediju datu plūsmas tendences, optimizēt tīkla konfigurāciju, uzlabot veiktspēju, samazināt drošības riskus un nodrošināt lietotājiem kvalitatīvākus video pakalpojumus.

1.2. Pamatojums

Lai nodrošinātu detalizētu interaktīvo mediju tīkla datu plūsmas uzraudzību un klasifikāciju, nepieciešama strukturēta daudzpakāpju metodoloģija. Pirmais būtiskais solis ir visaptverošas datu plūsmas apkopošana. Lai nodrošinātu daudzveidību un atbilstību, interaktīvo mediju datu plūsma tiek kategorizēta trīs galvenajās lietotņu jomās – tiešraides straumēšana (*Live Streaming*), mākoņspēles (*Cloud Gaming*) un metavide (*Metaverse*). Tiešraides straumēšanas kategorijā ietilpst datu plūsma no piecām globāli un reģionāli populārajām platformām – *YouTube, BiliBili, Twitch, Facebook* un *TikTok*. Mākoņspēļu kategorijā attiecināmas datu plūsmas, ko veido tādas lietotnes kā *Spitlings, Tomb Raider* un *Thumper* [98]. Metavides kategorijā ietilpst tādas lietotnes kā, piemēram, *VRChat, TheLab, SolarSystemAR, RealityMixer, Hellblade, DiRT Rally 2.0* un *Bigscreen Theatre* [108]. Visi datu plūsmas avoti ir iegūti gan no publiskām, gan patentētām (nepubliskām) datu kopām ar atšķirīgiem tīkla standartiem, apkopošanas ilgumu, datu ieguves rīkiem un ierīču platformām. Ir nepieciešama iegūto datu apstrāde, lai tos normalizētu un sagatavotu turpmākai analīzei.

Otrais solis – noteikt piemērotu datu plūsmas kategoriju skaitu un atbilstošu struktūru detalizētajai klasifikācijai. Galvenie faktori, kas ietekmē datu plūsmas raksturojumus, it īpaši video lietotnēm, ir izšķirtspēja, bitu pārraides ātrums (*bitrate*), kodēšanas formāts, konteineru formāts, kadru ātrums, straumēšanas režīms, piemēram, video pēc pieprasījuma (*Video on Demand – VoD*) vai tiešraides video straumēšana (*Video Live Streaming – VLS*), kā arī straumēšanas protokoli. Šie parametri un iepriekš nozarē iegūtās zināšanas palīdz definēt nozīmīgas un atšķiramas datu plūsmas kategorijas. Ja datu plūsma ir šifrēta, klasifikācijas detalizācijas līmenis būtiski ietekmē gan precizitāti, gan efektivitāti. Daudzi pētījumi koncentrējas uz datu plūsmas un datu vērtuma klasifikāciju, taču lielāka nozīme ir laika rindu

pieejām – tiek izmantotas datu plūsmas telpiskās un laika apgabala (spatiotemporālās) pazīmes bez nepieciešamības pārbaudīt datu lietderīgo daļu, šādi veicot klasifikāciju bez privātuma pārkāpšanas. Ceturtajā solī tiek veidotas datu kopas, balstoties izvēlētajos klasifikācijas detalizācijas līmeņos. Noslēdzošais solis ir atbilstošu apmācības pazīmju (*feature*) izvēle un uzraudzītas mācīšanās algoritmu lietošana klasifikācijas uzdevuma izpildei. Izmantojot standarta klasifikācijas metrikas, tiek novērtēta modeļa veiktspēja, lai izvērtētu detalizētas interaktīvo mediju tīkla datu plūsmas uzraudzības un klasifikācijas efektivitāti [13], [2], [3].

1.3. Promocijas darba mērķis un tēzes

Promocijas darba mērķis ir izstrādāt pieeju tīkla datu plūsmu detalizētajai uzraudzībai un klasifikācijai pēc to interaktīvo mediju lietotnēm, ņemot vērā dažādus tīkla apstākļus un detalizācijas līmeņus. Darbs paplašina izpratni par datu plūsmas modeļiem un interaktīvo mediju nozīmi mūsdienu sarežģītajās, heterogēnajās tīkla vidēs, vienlaikus veicinot datu plūsmas uzvedības analīzi tīkla datu plūsmas klasifikācijas (*NTC*) un tīkla datu plūsmas monitoringa un analīzes (*NTMA*) uzdevumos.

Promocijas darba aizstāvamās tēzes

1. tēze. Tīkla joslas platums ir galvenais faktors, kas ietekmē interaktīvo mediju tīkla datu plūsmas apjomu un sadalījumu. Nozīmīga ietekme ir arī video izšķirtspējai un kadru ātrumam, kas tieši ietekmē datu plūsmas raksturlielumus. Datu plūsmas raksturu būtiski ietekmē mainīga bitu ātruma (*Variable Bitrate – VBR*) kodēšana, atslēgas kadru (*keyframe*) bitu ātruma svārstības un *GOP* (*Group of Pictures*) struktūras.

2. tēze. Protokoli, ko izmanto straumēšanas lietotnēs, spēj dinamiski sadarboties ar dažādiem transporta slāņa protokoliem. *UDP* un *WebRTC* kombinācija ģenerē lielu pakešu skaitu ar mazu lietderīgo datu apjomu (vērtumu), bet datu plūsmas kopējais apjoms veido tikai 70 % no attiecīgā oriģinālā video faila izmēra, vienlaikus nodrošinot viszemāko laika aizturi (latentumu). Savukārt *TCP* un *HLS* apvienojums tādos pašos tīkla apstākļos veido vislielāko kopējo lietderīgo datu daudzumu, kas var sasniegt līdz 1,5 reizes no oriģinālā video faila izmēra, taču tādai kombinācijai raksturīga augsta attiecība starp vadības un datu plūsmu apjomiem.

3. tēze. Lietojumos ar detalizētu klasifikāciju patiesās klases ir tieši saistītas ar paraugu pazīmēm. Šīs pazīmes tiek veidotas, pamatojoties uz faktoriem, kas ietekmē datu plūsmas, kā arī no citām iepriekšējām zināšanām par datu plūsmu. Visas mācīšanās pazīmes un paraugi ir pārredzami un viegli interpretējami. Klasifikācijas uzdevumu veic trīs detalizācijas līmeņos – laika rindu, datu plūsmu un lietderīgo datu (vērtuma) līmeņos, pielāgojot pieeju dažādiem scenārijiem.

4. tēze. Slāņu modeļa (*Stacking model*) kopējā klasifikācijas precizitāte pārsniedz 99 % tiešraides straumēšanas datu kopai ar vairāk nekā 200 000 paraugiem 86 dažādās kategorijās. Datu plūsmas līmenī katrai no šīm kategorijām F1 rādītājs sasniedz vismaz 0,94. *1D-CNN* modeļa klasifikācijas precizitāte sasniedz vismaz 80 % gan laika rindas, gan datu plūsmas līmenī, tādējādi pārspējot trīs modernākus dziļās mācīšanās modeļus. Savukārt lietderīgo datu

līmenī *2D-CNN* modeļa klasifikācijas precizitāte pārsniedz 84 %, pārspējot citas pieejamās metodes par vairāk nekā 10 %.

Promocijas darba pētījuma hipotēzes

1. Vienāda tīkla joslas platuma apstākļos dažādas interaktīvo mediju lietotnes ģenerē līdzīgā apjoma datu plūsmas. Neapstrādātā video materiāla parametru variācijas nosaka ne vairāk kā 10 % atšķirību kopējā plūsmas apjomā starp visām lietotnēm. Pieņemot *GOP* vērtību vienādu ar 180 kadriem, ir iespējams visprecīzāk atspoguļot faktisko datu plūsmas sprādzienvieda sadalījumu laikā.

2. *WebRTC* un *UDP* kombinācija pārraida vismazāko lietderīgo datu apjomu, ļaujot ietaupīt vairāk nekā 30 % datu, salīdzinot ar sākotnējā video faila izmēru. Šāda kombinācija ģenerē vismaz divreiz vairāk pakešu nekā citi straumēšanas protokoli, tostarp *HLS*, *DASH*, *RTMP* un *HTTP-FLV*.

3. Detalizētas klasifikācijas iezīmes korelē ar zināmajiem datu plūsmas ģenerēšanas parametriem, nodrošinot pārredzamu un izskaidrojamu klašu sasaisti. Datu vērtuma līmenī interaktīvo mediju datu plūsmas paraugu kopējais skaits pārsniedz tādu laika rindas un datu plūsmas līmeņos par vismaz 70 %, saglabājot paraugu noņemšanas biežumu.

4. *2D-CNN* modeļa klasifikācijas precizitāte sasniedz vairāk nekā 84 % datu vērtuma līmenī, pārspējot citus salīdzinātos modeļus par vismaz 10 %.

1.4. Promocijas darba uzdevumi

1. Apkopoti interaktīvo mediju lietotņu tīkla plūsmas dati, kas tika savākti no vairākiem avotiem, tostarp publiskajām un privātām datu kopām. Datu plūsmas tika aktīvi uzkrātas no dažādām platformām, lietotnēm un pakalpojumiem dažādos tīkla apstākļos. Divu virzienu datu plūsma no klienta un servera puses tika iegūta idealizētā tīkla vidē, izmantojot vairākus straumēšanas protokolus.

2. Klienta-servera "*push*" un "*pull*" straumēšanas testiem tika izmantots standarta fails video kvalitātes pārbaudei – "*Big Buck Bunny*". Izveidoti video faili ar dažādiem kvalitātes iestatījumiem, izmantojot video kodēšanas rīku "*FFmpeg*". Tika izpētītas likumsakarības starp bitu ātrumu un kadru sadalījumu.

3. Veikta globāla un lokāla statistiskā analīze interaktīvo mediju tīkla datu plūsmām. Izmantojot slīdošā laika loga metodi, tika analizēts datu plūsmas sadalījums dažādos laika mērogos, kas ļauj identificēt modeļus un to raksturus.

4. Izstrādātas algoritmu metodes, kas ļauj ģenerēt klasifikācijas datu kopas interaktīvo mediju lietotņu tīkla datu plūsmām visos trīs līmeņos – laika rindas, plūsmas un lietderīgo datu līmenī. Veikta datu pirmapstrāde atbilstošo datu kopu veidošanai.

5. Izveidoti un novērtēti klasifikācijas modeļi, tostarp *1D-CNN*, *2D-CNN*, *RNN*, *GCN* un slāņu modelis. To veiktspēja tika salīdzināta ar trim mūsdienīgākajiem modeļiem – *Deep Packet*, *FlowPic* un *FS-Net*.

1.5. Pētījuma metodes

Lai īstenotu promocijas darba uzdevumus un analizētu ar tiem saistītās problēmas, tīklu datu plūsmu pārtveršanai tika izmantoti tādi rīki kā *Wireshark*, *Tcpdump* un *nDPI*. Datu pirmapstrāde tika veikta, izmantojot *MATLAB*, *NFStream* un *CICFlowMeter*, veidojot īpašumtiesību datu kopas, kas tika salīdzinātas ar dažādām publiski pieejamām references datu kopām. Matemātiskajai modelēšanai tika lietoti vairāki mūsdienīgie mašīnmācīšanās algoritmi no populārajām *Python* bibliotēkām (*PyTorch*, *PyTorch Geometric*, *OpenCV*, *SciPy*, *StellarGraph*, *TensorFlow*, *Keras*, *XGBoost*, *scikit-image* un *scikit-learn*) ar mērķi veikt tīkla datu plūsmas klasifikāciju. Papildus tam tika izmantoti 5G tīklu rīki, tādi kā *srsRAN*, *Simu5G* un *SRS*, kas ļauj simulēt tīkla datu plūsmas specifiskajos scenārijos.

Promocijas darba galvenie eksperimenti tika veikti "*Open RAN* testēšanas un izpētes laboratorijā", Rīgas Tehniskās universitātes (RTU) Fotonikas, elektronikas un elektronisko sakaru institūtā (FEESI). Papildu eksperimentālais darbs tika veikts Kartahenas Tehniskās universitātes (*UPCT*; Spānija) Informācijas un komunikācijas tehnoloģiju katedrā.

1.6. Zinātniskā novitāte un galvenie rezultāti

Promocijas darba zinātniskā novitāte izpaužas kā novatoriska pieeja interaktīvo mediju lietotņu tīkla datu plūsmu detalizētās uzraudzības un klasifikācijas veikšanai heterogēnajos tīklos. Pētījumā tika sasniegta augsta klasifikācijas precizitāte pēc vairākiem rādītājiem, izmantojot pārredzamu un interpretējamu analīzes procesu. Tālāk uzskaitīti konkrēti sasniegumi, kas atspoguļoti astoņās zinātniskajās publikācijās.

1. Izstrādāts robusts slāņu ansambļa modeļa (*ensemble*) klasifikators, kura precizitāte sasniedza 99 %, klasificējot 38 kategorijas, un F1 makro rādītājs 0,83, klasificējot 101 kategoriju plūsmas līmenī, izmantojot gan publiskās, gan privātās datu kopas. Šāds modelis efektīvi risina problēmas, kas saistītas ar nevienmērīgu kategoriju sadalījumu un dažādiem paraugu apjomiem. Slāņu modelis, kas aprakstīts 1. publikācijā, tiek izmantots metodoloģijas nodaļā, klasifikācijas modeļu apakšnodaļā.

2. Demonstrēta 5G frekvenču joslas un joslas platuma ietekme uz video lietotņu datu plūsmas sadalījumu 5G *Open RAN* testa vidē. Izmantojot uzlaboto *Mobile BroadBand Slicing* scenāriju, plūsmas līmenī 13 kategorijās tika sasniegta klasifikācijas precizitāte, kas pārsniedz 99 %. Attiecīgās publiskās datu kopas, kas aprakstītas 2. publikācijā, tiek izmantotas metodoloģijas nodaļas datu iegūšanas un pirmapstrādes apakšnodaļās.

3. Izstrādāts jauns un izskaidrojams *nYFTQC* algoritms, kas nodrošina 15 privāto interneta video plūsmu datu kopas ar augstu pilnīguma, pēctecības un pārredzamības pakāpi. Inovatīvais *nYFTQC* algoritms, kas aprakstīts 3. publikācijā, ir lietots metodoloģijas nodaļas plūsmas līmeņa klasifikācijas datu kopu ģenerēšanas apakšnodaļā.

4. Izmantojot grafu konvolūcijas tīkla (*Graph Convolutional Network – GCN*) dziļās mācīšanās modeli, tika sasniegta 86 % klasifikācijas precizitāte 10 kategorijās plūsmas līmenī, kur tika analizētas gan reālu *OTT* lietotņu datu plūsmas, gan *DASH* simulētas video datu plūsmas. *GCN* modelis aprakstīts 4. publikācijā, tā veiktspējas analīzes procedūra aprakstīta

metodoloģijas nodaļas klasifikācijas modeļu apakšnodaļā. 4. publikācijā aprakstīta arī *DASH* video straumēšanas simulācijas metode, kas tika izmantota datu iegūšanas un pirmapstrādes apakšnodaļā simulēto *VLS* datu plūsmas ģenerēšanai.

5. 5. publikācijā izstrādāts uzlabotais *GNN* modelis, ar kura palīdzību tika veikta klasifikācija *OTT* un paplašinātās realitātes (*XR*) jaukto datu plūsmu kopai. Paraugi tika marķēti trīs kategorijās, balstoties lietotāju kvalitātes (*QoE*) vērtējumā pēc vidējā viedokļa vērtējuma (*Mean Opinion Score – MOS*) metrikas. Sasniegta aptuveni 90 % klasifikācijas precizitāte pakešu līmenī *Wi-Fi* tīkla kanālā. 5. publikācijā jauktā datu plūsmas ģenerēšanas metode tika izmantota metodoloģijas nodaļā datu iegūšanas un pirmapstrādes apakšnodaļā metavides un mākoņspēļu datu kopu izveidei. Pakešu līmeņa klasifikācijas metode, kuras pamatā ir baitu pakešu grafu paraugu veidošana, tika izmantota metodoloģijas nodaļas vairāku kategoriju detalizētās datu kopas ģenerēšanas apakšnodaļā ar mērķi izveidot pakešu līmeņa interaktīvo mediju plūsmas datu kopas.

6. Lietots ģeneratīvs *DBN* (*Deep Belief Network*) modelis, kura klasifikācijas precizitāte sasniedza 92 % laika rindas līmenī astoņās kategorijās, analizējot normālu *OTT* lietotņu datu plūsmu *Wi-Fi* tīklā, kas tika sajaukta ar *Tor* anonimizācijas un *VPN* šifrēšanas datu plūsmām. Tāds pats modelis plūsmas līmenī uzrādīja 94 % precizitāti šai pašai datu kopai. 6. publikācijā izmantotas publiskas datu kopas ar *Tor* un *VPN*, kurās pēc to pirmapstrādes tika iekļautas metodoloģijas nodaļas datu iegūšanas un pirmapstrādes apakšnodaļā, lai izveidotu jaunu *Tor&VPN* datu apakškopu kā tiešraides straumēšanas datu kopas daļu.

7. Lietots *1D-CNN* modelis, kura precizitāte sasniedza 97 %, klasificējot *YouTube* četrās kategorijās laika rindas līmenī. Uz šī modeļa pamata tika izstrādāti arī modificēti *2D-CNN* un *RNN* modeļi, kas spēja paplašināt to lietošanas iespējas arī citos detalizācijas līmeņos. 7. publikācijā aprakstītais *1D-CNN* modelis tika izmantots metodoloģijas nodaļas klasifikācijas modeļu apakšnodaļā, lai veiktu klasifikācijas veiktspējas metriku salīdzināšanu.

8. Tika lietota statistisko momentu analīze un ātrā Furjē transformācija (*FFT*), lai risinātu pakešu sprādzienrakstura un sadalījuma nobīdes problēmas laika rindas līmenī. 8. publikācijā aprakstītās metodes uzlaboja rezultātu interpretēšanu, bet nodrošināja tikai ierobežotu klasifikācijas precizitātes uzlabojumu. Šīs metodes tika izmantotas, lai pārbaudītu interaktīvo mediju tīkla datu plūsmas klasifikācijas veiktspēju laika rindas līmenī, kā tas ir atspoguļots metodoloģijas nodaļas vairāku kategoriju detalizācijas datu kopas ģenerēšanas apakšnodaļā.

Pamatojoties uz klasifikācijas rezultātiem, trīs detalizācijas līmeņos datu kopām, kas satur interaktīvo mediju lietotņu tīkla datu plūsmu, tika iegūti šādi galvenie rezultāti.

1. *1D-CNN* modelis sasniedza līdz pat 95 % klasifikācijas precizitāti un makro F1 rezultātu 0,93, klasificējot 95 kategorijas pakešu laika rindas līmenī tiešraides straumēšanas datu kopā ar vairāk nekā 40 000 paraugiem.

2. Slāņu modelis sasniedza līdz pat 99 % klasifikācijas precizitāti un makro F1 rezultātu 0,99, klasificējot 86 kategorijas plūsmas līmenī tiešraides straumēšanas datu kopā ar vairāk nekā 200 000 paraugiem.

3. 2D-CNN modelis sasniedza līdz pat 81 % klasifikācijas precizitāti un makro F1 rezultātu 0,82, klasificējot 95 kategorijas pakešu līmenī tiešraides straumēšanas datu kopā ar vairāk nekā 180 000 paraugiem.

Promocijas darba izstrādes gaitā iegūtie rezultāti izmantoti vairākos projektos.

1. Eiropas Reģionālās attīstības fonda (ERAF) projekts darbības 1.1.1.2. "Pēcdoktorantūras pētniecības atbalsts" ietvaros, specifiskā atbalsta mērķa 1.1.1 ietvaros, Nr. 1.1.1.2/VIAA/2/18/332, 01.09.2021.–30.11.2021.

2. Eiropas Sociālā fonda (ESF) projekts "Doktorantu un akadēmiskā personāla specializācijas stratēģiskā stiprināšana", Nr. 8.2.2.0/20/I/008, 01.12.2022.–30.11.2023.

3. Eiropas Savienības Atveseļošanas un noturības mehānisma (*RRF*) projekts, Nr. 5.2.1.1.i.0/2/24/I/CFLA/003, 01.10.2024.–30.09.2025.

4. *Open RAN* projekts "*Multipath Roaming Solution for Open RAN* (*TRANSFER*)", Nr. 1.1.1.3/1/24/A/018, 24.04.2025.–30.09.2025.

5. Spānijas projekts ar dotāciju *PID2023-148214OB-C21*, finansējums – *MICIU/AEI/10.13039/501100011033* un *FEDER/EU* (*Mursija, Spānija*), 01.05.2023.–31.12.2024.

1.7. Promocijas darba struktūra

Promocijas darbu veido četras galvenās nodaļas. 1. nodaļā aplūkots pētījuma konteksts par interaktīvo mediju tīkla datu plūsmu detalizētu uzraudzību un klasifikāciju. Tajā izklāstīts pētījuma konteksts, motivācija, literatūras apskats, formulēti pētījuma mērķi, uzdevumi, tēzes un hipotēzes. 2. nodaļā aprakstītas metodoloģijas, kas tika lietotas definēto pētniecības mērķu sasniegšanai. Nodaļai ir piecas apakšnodaļas: datu savākšana un pirmapstrāde; datu plūsmas modeļu analīze; vairāku kategoriju detalizētās klasifikācijas datu kopas ģenerēšana; modeļu izveide; eksperimentālā vide. 3. nodaļā sniegta detalizēta klasifikācijas rezultātu analīze trīs detalizācijas līmeņos, kā arī atbilstošo datu kopu vērtējums. 4. nodaļā apkopti promocijas darba rezultāti un iespējamie nākotnes pētījumu virzieni.

1.8. Promocijas darba publikācijas un aprobācija

Promocijas darba rezultāti publicēti **piecos** zinātniskajos rakstos un konferenču tēžu krājumos, no kuriem **četri** ir iekļauti *SCOPUS, Web of Science* (*WoS*) un *Institute of Electrical and Electronics Engineers* (*IEEE*) datubāzēs. Autoram ir **astoņas** publikācijas. Galvenie promocijas darba rezultāti apkopoti **trīs** zinātniskajos žurnālos. Pētījuma rezultāti prezentēti **astoņās** starptautiskajās zinātniskajās konferencēs.

Zinātniskās publikācijas, kurās atspoguļoti promocijas darba rezultāti

1. **Chen, T.**, Grabs, E., Ipatovs, A., Cano M. "Ensemble Learning Enabled Flow-level Internet Traffic Classification" – iesniegts publicēšanai žurnālā "*Journal of Information and Telecommunication*" (izskatīšanā).

2. **Chen, T.**, Benkis R., Bogdanovs, N., Stetjuha M., Klūga J., Jeralovičs V., Rjazanovs D., Grabs, E., Ipatovs, A. "Video Traffic Classification in the 5G Open RAN Network" – pieņemts publicēšanai **2025**. gada Photonics & Electromagnetics Research Symposium (PIERS 2025) konferencē, Apvienotie Arābu Emirāti, Abū Dabī, 2025. gada 4.–8. maijs.

3. **Chen, T.**, Grabs, E., Ipatovs, A., Cano, M. "A Novel Proprietary Internet Video Traffic Dataset Generation Algorithm". Applied Sciences, **2025**, Vol. 15, No. 2, Raksta Nr. 515. e-ISSN 2076-3417.

4. **Chen, T.**, Grabs, E., Ipatovs, A., Pētersons, E., Ancāns, A. "Bitrate-based Video Traffic Classification". In: 2023 Photonics & Electromagnetics Research Symposium (PIERS 2023): Proceedings, Čehija, Prāga, **2023**. gada 3.–6. jūlijs.
5. **Chen, T.**, Grabs, E., Tasic, I., Cano, M. "Network Traffic Analysis for eXtended Reality Applications". In: XVI Telematics Engineering Conference (JITEL 2023), Spānija, Barselona, **2023**. gada 8.–10. novembris.
6. **Chen, T**., Grabs, E., Pētersons, E., Efrosinin, D., Ipatovs, A., Bogdanovs, N., Rjazanovs, D. "Multiclass Live Streaming Video Quality Classification Based on Convolutional Neural Networks". Automatic Control and Computer Sciences, **2022**, Vol. 54, No. 5, 455–466. ISSN 0146-4116, e-ISSN 1558-108X.
7. **Chen, T.**, Grabs, E., Pētersons, E., Efrosinin, D., Ipatovs, A., Klūga, J. "Encapsulated and Anonymized Network Video Traffic Classification with Generative Models". In: Workshop on Microwave Theory and Techniques in Wireless Communications (MTTW'22), Latvija, Rīga, **2022**. gada 5.–7. oktobris.
8. Grabs, E., **Chen, T.**, Pētersons, E., Efrosinin, D., Ipatovs, A., Klūga, J., Čulkovs, D. "Features Extraction for Live Streaming Video Classification with Deep and Convolutional Neural Networks". In: IEEE Microwave Theory and Techniques in Wireless Communications (MTTW 2021), Latvija, Rīga, **2021**. gada 7.–8. oktobris.

Starptautiskās zinātniskās konferences, kurās prezentēti promocijas darba rezultāti
1. **Chen, T.**, Benkis R., Bogdanovs, N., Stetjuha M., Klūga J., Jeralovičs V., Rjazanovs D., Grabs, E., Ipatovs, A. "Video Traffic Classification in the 5G Open RAN Network" – PIERS 2025, Apvienotie Arābu Emirāti, Abū Dabī, **2025**. gada 4.–8. maijs.
2. **Chen, T.**, Grabs, E., Tasic, I., Cano, M. "Network Traffic Analysis for eXtended Reality Applications". JITEL 2023, Spānija, Barselona, **2023**. gada 8.–10. novembris.
3. **Chen, T.**, Grabs, E., Ipatovs, A. "Artificial Intelligence Application in Network Traffic Engineering", RTU 63. starptautiskā zinātniskā konference, Latvija, Rīga, **2023**. gada 17. oktobris.
4. **Chen, T.**, Grabs, E., Ipatovs, A., Pētersons, E., Ancāns, A. "Bitrate-based Video Traffic Classification". PIERS 2023, Čehija, Prāga, **2023**. gada 3.–6. jūlijs.
5. **Chen, T.**, Grabs, E., Ipatovs, A. "Artificial Intelligence Application in Network Traffic Engineering", RTU 64. starptautiskā zinātniskā konference, Latvija, Rīga, 2022. gada 14. oktobris.
6. **Chen, T.**, Grabs, E., Pētersons, E., Efrosinin, D., Ipatovs, A., Klūga, J. "Encapsulated and Anonymized Network Video Traffic Classification with Generative Models". MTTW'22, Latvija, Rīga, **2022**. gada 5.–7. oktobris.
7. Grabs, E., **Chen, T.**, Ipatovs, A. "Flow-based Video Quality Traffic Classification for Real-Time SDN Applications", ERI Workshop on Telecommunication and Networks, Rumānija, Kluža-Napoka, **2022**. gada 14.–15. marts.
8. Grabs, E., **Chen, T.**, Pētersons, E., Efrosinin, D., Ipatovs, A., Klūga, J., Čulkovs, D. "Features Extraction for Live Streaming Video Classification with Deep and Convolutional Neural Networks". MTTW 2021, Latvija, Rīga, **2021**. gada 7.–8. oktobris.

2. METODOLOĢIJA

Metodoloģijas nodaļā visi izmantotie paņēmieni tiek sistemātiski izklāstīti piecās apaknodaļās: datu iegūšana un pirmapstrāde; interaktīvās multivides tīkla datu plūsmas raksturojumu analīze; vairāku kategoriju klasifikācijas datu kopu ģenerēšana; klasifikācijas modeļi; eksperimentālā konfigurācija.

2.1. Datu iegūšana un pirmapstrāde

Datu iegūšana un pirmapstrāde ir būtiskie soļi, ko jāizpilda standartizētas apmācības datu kopas izveidošanai ar mērķi veikt interaktīvo mediju lietotņu tīkla datu plūsmas analīzi. 1. tabulā apkopota informācija par tīkla datu plūsmām interaktīvo mediju lietotnēs, ko veido četri datu kopumi – tiešraides straumēšana, mākoņspēles, metavide un tiešsaistes video straumēšanas (*VLS*) simulācija. Katrs datu kopums ietver vairākas datu apakškopas. Tiešraides straumēšanas datu kopā katra īpašumtiesību datu apakškopa atbilst konkrētai *OTT* platformai. Datu plūsmas dati tika iegūti, atskaņojot vienu video katrā platformā, izmantojot *Google Chrome* versiju *100.0.4896.88*. Datu plūsma tika pārtverta *Packet Capture* (*PCAP*) failu veidā, izmantojot *Wireshark 3.4.3 Windows 10* sistēmā (*Build 18363.1316*), ar *Intel(R) Wi-Fi 6 AX200* 160 MHz adapteri monitoringa režīmā. Iegūtie *PCAP* faili tika pārbaudīti un filtrēti pēc interneta protokola (*IP*) un *Media Access Control* (*MAC*) adresēm, kā arī ierobežoti garumā līdz 30 minūtēm, lai nodrošinātu atbilstošu procesa ilgumu visos izmantotajos pakalpojumos. Datu apakškopas sadalītas kategorijās pēc platformas nosaukuma, video izšķirtspējas, kadru skaita sekundē (*FPS*) un satura veida – video pēc pieprasījuma vai tiešraides video straumēšanas. Apkopoto datu plūsmu kopējais apjoms pārsniedz 84 GB datu.

YouTube datu apakškopā tika novērota vislielākā kategoriju daudzveidība. Dažādas *OTT* platformas rekomendē atšķirīgus video kodēšanas iestatījumus. Lielākā daļa no platformām izmanto adaptīva bitu pārraides ātruma (*ABR*) straumēšanu, lai dinamiskajā veidā pielāgotu video kvalitāti izejot no tīkla stāvokļa un ierīces jaudas [109], [20]. Datu pārtveršanas laikā tika izvēlēta fiksēta atskaņošanas izšķirtspēja un latentuma režīms, savienojuma datu pārraides ātrums *WiFi* gadījumā pārsniedza 100 Mbps. Tomēr daudzas platformas, piemēram, *YouTube* un *Vimeo*, dod priekšroku kodēšanai ar mainīgo bitu pārraides ātrumu (*VBR*), kas ļauj bitu pārraides ātrumam svārstīties, nodrošinot optimālu kvalitāti [110]–[112]. *VBR* kodēšanas lietošanu ietekmē vairāki faktori, tostarp izšķirtspējas parametri, kadru skaits sekundē, krāsu dziļums, kodeks, kodēšanas iestatījumi, ainavas sarežģītība un kustības intensitāte. Palielinoties kādam no šiem parametriem, parasti pieaug arī nepieciešamais bitu pārraides ātrums. Savukārt tādas platformas kā *Twitch*, *Facebook*, *TikTok* un *BiliBili* bieži dod priekšroku kodēšanai ar nemainīgo bitu pārraides ātrumu (*CBR*) [113]–[115]. *VoD* lietotnēm lejupielādētos video failus analizē, lai iegūtu neapstrādātu bitu pārraides ātruma informāciju. Savukārt *VLS* gadījumā šādi dati jānosaka netiešā veidā, izmantojot reālā laika datu plūsmas statistiku. Šādu pieeju ietekmē tādi faktori kā satura piegādes tīkla (*CDN*) kešatmiņas aiztures, video resursu glabāšanas vieta un mainīgie tīkla apstākļi, tostarp joslas platums, trīce (*jitter*) un pakešu zudumu varbūtība [116].

5G tīkla kontekstā datu plūsmas raksturojums ir būtiski mainījies dažādu mobilo tīklu konfigurāciju variāciju dēļ. Saskaņā ar "5G pārklājuma paplašināšanas datu kopu 1", "Itālijas laboratorijas *testbed* datu kopu 1" un "Itālijas laboratorijas testbed datu kopu 2" no *NANCY* projekta video datu plūsma tika simulēta un apkopota, izmantojot *srsRAN* un *PCAPdroid* rīkus *Open RAN* tīkla vidē. Dati tika iegūti no vairākiem protokola slāņiem, tostarp *MAC*, nākamās paaudzes lietotņu protokola (*NGAP*), *Radio Link Control* (*RLC*) un *GPRS* tunelēšanas protokola lietotāja plāna (*GTP-U*), kā arī perifērdatošanas (*edge*) klienta pusē. Dažādi video faili tika straumēti, izmantojot atšķirīgo izšķirtspēju (480 P, 720 P, 1080 P, 2160 P un 4320 P), vairākas bitu pārraides ātruma vērtības (6 Mbps, 10 Mbps, 20 Mbps un 40 Mbps) un dažādus joslas platumus (10 MHz un 20 MHz), izmantojot divus frekvenču joslu diapazonu (N3 un N78) ar reāllaika ziņojumapmaiņas protokolu (*RTMP*). Rezultātā iegūto nemainīgo bitu pārraides ātrumu saraksts tiek aproksimēts, izmantojot ātruma ierobežotāju dažādu tīkla joslas platuma apstākļos. Tomēr teorētiskais maksimālais pārraides ātrums 5G tīklos būtiski pārsniedz bitu pārraides ātrumu. Kodēšana tika veikta, izmantojot *H.264* un *VP8* kodekus. Šie datu plūsmas ieraksti ir apvienoti *Big Buck Bunny* apakšdatu kopā, tiešraides straumēšanas kategorijā [117]–[119].

Turklāt pat vienā un tajā pašā tīkla vidē datu plūsmas sadalījums var mainīties atkarībā no dažādiem *ABR* protokoliem. Šo variāciju ietekmē atšķirības pakešu kešdarbes procedūrā, latentuma tolerancē, ierīču jaudās un lietotāju uzvedībā. Tādas platformas kā *Twitch*, *Facebook* un *TikTok* galvenokārt izmanto *HTTP Live Streaming* (*HLS*) protokolu. Savukārt *BiliBili* un *Netflix* galvenokārt lieto *Dynamic Adaptive Streaming over HTTP* (*DASH*), savukārt *YouTube* un *Vimeo* atbalsta abus protokolus. *HLS*, kas tika definēts atbilstoši *RFC 8216*, atbalsta segmentētu video pārraidi. Klienti saņem un atskaņo video saturu, izmantojot *.m3u8* atskaņošanas sarakstu failus un *.ts* segmentu failus [120]. Parasti tiek izmantots *H.264* video kodeks un tiek atbalstīti audio kodeki, piemēram, *AAC*, *MP3* un *AC-3*. *DASH*, standartizēts saskaņā ar *ISO/IEC 23009-1:2022*, arī atbalsta segmentētu pārraidi. Klienti izmanto *Media Presentation Description* (*MPD*) atskaņošanas sarakstu, lai piekļūtu *.ts* vai *.m4s* segmentu failiem. *DASH* atbalsta plašāku video kodeku klāstu, tostarp *H.265*, *H.264* un *VP9* [121]. Lai gan daudzi video avota parametri (piemēram, kodēšanas formāts, izšķirtspēja, bitu pārraides ātrums, kadru ātrums, protokols) var tikt pielāgoti pirms straumēšanas, šīs detaļas parasti klientam nav redzamas. Katram atskaņošanas gadījumam *OTT* platformā ir unikāls datu plūsmas "pirkstu nospiedums", kura pamatā ir slēptie parametri. Tādējādi datu plūsmas raksturojums interaktīvo mediju lietotnēm ievērojami atšķiras atkarībā no izvēlētā protokola, platformas un atskaņošanas scenārija.

Lai novērstu ārējo vides parametru ietekmi uz tīkla datu plūsmas sadalījumu interaktīvo mediju lietotnēs, tiek sistemātiski pētītas datu plūsmas īpašības dažādu straumēšanas protokolu gadījumā. Visi serveri un klienti tiek konfigurēti vienā lokālā tīkla (*LAN*) vidē, lai nodrošinātu piemērotus testēšanas apstākļus. Viens un tas pats *Big Buck Bunny* video fails tiek atkārtoti izmantots vairākās *VLS* sesijās, lietojot dažādus *ABR* protokolus. Katras sesijas laikā lejupielādes datu plūsma tiek uztverta klienta pusē, savukārt augšupielādes datu plūsma tiek pārtverta *Simple Realtime Server* (*SRS*) serverī, izmantojot *Tcpdump* rīku [122]. Divvirzienu datu plūsma tiek izvēlēta kā atsevišķa kategorija. *Big Buck Bunny* tiek uzskatīts par plaši

pieņemtu etalonu video atskaņošanas pārbaudei un izpētei, pateicoties augstas kvalitātes neapstrādāto video formātu pieejamībai, kas nodrošina elastīgu pirmapstrādi eksperimentu vajadzībām [123]. *SRS* serveris darbojas *Docker* vidē un atbalsta vairākus straumēšanas protokolus, tostarp *RTMP*, *Web Real-Time Communication* (*WebRTC*), *HLS*, *HTTP FLV* tiešraidi (*HTTP-FLV*), *Secure Reliable Transport* (*SRT*), *DASH* un *GB28181*. Šim pētījumam tika izvēlēti pieci protokoli, kas ir būtiski interaktīvo mediju lietotnēm: *RTMP*, *WebRTC*, *HLS*, *HTTP-FLV* un *DASH*. *RTMP* tiek izmantots kā straumēšanas pārraides protokols, bet video rediģēšanai, transkodēšanai un segmentu ģenerēšanai tiek izmantots *FFmpeg* [124]. Katram multivides segmentam ir 2 sekunžu garums, un katrā slīdošā loga sesijā tiek saglabāti 10 segmenti. Visus piecus izvēlētos protokolus klienta pusē atbalsta atskaņošanas rīki, piemēram, *SRS Player*, *VLC*, *Dash JavaScript Player* un *FFplay* [125], [126]. No šīm straumēšanas sesijām iegūtā datu plūsma tiek apkopota tiešraides video straumēšanas simulācijas (*Streaming VLS Simulation*) datu kopā, kuras kopējais apjoms ir aptuveni 20 GB.

Mākoņspēļu un metavides datu kopām ir būtiskas atšķirības datu raksturlielumos. Mākoņspēļu datu kopa ietver trīs apakškopas: *Spitlings* (*SP*), *Tomb Raider* (*TR*) un *Thumper* (*TH*), kas visas tika ņemtas *Google Stadia* platformā. *Stadia* pakalpojumi tiek nodrošināti, izmantojot *WebRTC* ietvaru, kas izmanto vairākus protokolus, tostarp *ICE*, *STUN*, *TURN*, *DTLS*, *RTP* un *RTCP* [127]. *ICE* (*Interactive Connectivity Establishment*) nodrošina vienādranga tīkla savienojumu virs *UDP* protokola *NAT* vidē, iekļaujot *STUN* un *TURN*. *DTLS* (*Datagram Transport Layer Security*) nodrošina drošu datagrammu pārraidi. *RTP* (*Real-Time Protocol*) tiek izmantots multivides piegādei no servera, savukārt *RTCP* (*Real-Time Control Protocol*) ļauj klientam sniegt atgriezenisko saiti par atskaņošanas kvalitāti. Tīkla datu plūsma tika pārtverta *VBR* režīmā bez joslas platuma ierobežojumiem. Katrai apakškopai piešķirta kategorija saskaņā ar spēles nosaukumu, izšķirtspēju, kodeku, spēles stāvokli un straumēšanas protokolu. Ņemot vērā to, ka neapstrādāti *PCAP* faili nav pieejami, detalizēta plūsmas un vērtuma līmeņa analīze nav iespējama [98].

Metavides datu kopas veido septiņi atšķirīgi pakalpojumi, un tīkla datu plūsma tika apkopota, izmantojot virtuālā datora straumēšanas (*VDS*) konfigurāciju. Mākoņdatošanas dators kalpoja kā apstrādes platforma, kurā darbojās *VDS* serveris. *Oculus Quest 2* ierīce tika savienota ar *VDS* klientu, un datu plūsma tika pārtverta, izmantojot *Wireshark*. Katras sesijas joslas platuma ierobežojums veidoja 15 Mbps, un tādas sesijas ilgums veidoja aptuveni vienu minūti. Līdzīgi mākoņspēļu datu kopai, arī metavides lietotnes balstās *WebRTC* protokolā straumēšanai [108]. Papildus ar video saistītajiem metadatiem *OTT* lietotnes bieži realizē anonimizācijas un šifrēšanas paņēmienus. Lai atspoguļotu tādas īpašības, *YouTube* un *Vimeo* datu plūsmas paraugus, kuros apvienota *Tor* anonimizācijas pārlūkprogramma un *OpenVPN* protokola iekapsulēšanas šifrēšana, dati iegūti no publiskajām *ISCXVPN2016* un *ISCXTor2016* datu kopām [29], [30]. Rezultējošā datu kopa ietver četras kategorijas, tomēr tajā trūkst detalizētu metadatu aprakstu par video parametriem. Noslēgumā tika analizētas interaktīvo mediju lietotņu tīkla plūsmas, ņemot vērā protokolu daudzveidību, video metadatus un tīkla vides raksturlielumus.

Tabula 1.

Interaktīvo mediju lietotņu tīkla datu plūsmas raksturlielumu variāciju kopsavilkums

Datu kopas	Datu kopa	Kategorija	Neapstrādāto *PCAP* failu izmērs	Ilgums	Tīkls	Avots
Tiešraides straumēšana	*YouTube*	26	44,8 GB	1800 s	*WiFi*	Patentēts
	Big_Buck_Bunny	13	10,4 GB	600 s	5G	Publisks [117], [119]
	BiliBili	15	20,2 GB	1800 s	*WiFi*	Patentēts
	Twitch	14	8,48 GB	1800 s	*WiFi*	Patentēts
	Facebook	12	3,78 GB	1800 s	*WiFi*	Patentēts
	TikTok	6	2,98 GB	1800 s	*WiFi*	Patentēts
	Vimeo	5	1,84 GB	1800 s	*WiFi*	Patentēts
	Tor&VPN	4	1,95 GB	945 s	*WiFi*	Publisks [29] [30]
Mākoņspēles	*Spitlings*	6	/	600 s	*WiFi*	Publisks [98]
	Tomb Raider	5	/	600 s	*WiFi*	Publisks [98]
	Thumper	1	/	600 s	*WiFi*	Publisks [98]
Metavide	*VRChat*	3	338 MB	63 s	*WiFi*	Publisks [108]
	TheLab	3	348 MB	63 s	*WiFi*	Publisks [108]
	SolarSystemAR	3	345 MB	63 s	*WiFi*	Publisks [108]
	RealityMixer	3	350 MB	63 s	*WiFi*	Publisks [108]
	Hellblade	3	347 MB	63 s	*WiFi*	Publisks [108]
	DiRTRally2.0	3	358 MB	63 s	*WiFi*	Publisks [108]
	BigscreenTheatre	3	357 MB	63 s	*WiFi*	Publisks [108]
Tiešraides video straumēšanas simulācija	*Big_Buck_Bunny*	13	4,14 GB	635 s	*Ethernet*	Patentēts
	Big_Buck_Bunny	13	3,23 GB	635 s	*Ethernet*	Patentēts
	Big_Buck_Bunny	13	3,17 GB	635 s	*Ethernet*	Patentēts
	Big_Buck_Bunny	13	3,22 GB	635 s	*Ethernet*	Patentēts
	Big_Buck_Bunny	13	2,81 GB	635 s	*Ethernet*	Patentēts
	Big_Buck_Bunny	13	3,23 GB	635 s	*Ethernet*	Patentēts

Datu kopas	Datu kopa	Izšķirtspēja	Kadru skaits sekundē	Straumēšanas protokols	Avota bitu ātrums
Tiešraides straumēšana	*YouTube*	144 P–8 K	30;60	*HLS/DASH*	*VBR/CBR*
	Big_Buck_Bunny	480 P–8 K	/	*HLS/DASH*	6; 10; 20; 40 Mbps
	BiliBili	360 P–8 K	30; 60; 80; 120	*DASH*	*VBR/CBR*
	Twitch	160 P–1080 P	30; 60	*HLS*	*VBR/CBR*
	Facebook	144 P–1080 P	30	*HLS*	*VBR/CBR*
	TikTok	360 P–1080 P	30; 60	*HLS*	*VBR/CBR*
	Vimeo	/	/	*HLS/DASH*	*VBR/CBR*
	Tor&VPN	/	/	*HLS/DASH*	VBR/CBR
Mākoņspēles	*Spitlings*	720 P–4 K	/	*WebRTC*	*VBR*
	Tomb Raider	720 P–4 K	/	*WebRTC*	*VBR*
	Thumper	1080 P	/	*WebRTC*	*VBR*
Metavide	*VRChat*	/	60; 90; 120	*WebRTC*	15 Mbps
	TheLab	/	60; 90; 120	*WebRTC*	15 Mbps
	SolarSystemAR	/	60; 90; 120	*WebRTC*	15 Mbps
	RealityMixer	/	60; 90; 120	*WebRTC*	15 Mbps
	Hellblade	/	60; 90; 120	*WebRTC*	15 Mbps
	DiRTRally2.0	/	60; 90; 120	*WebRTC*	15 Mbps
	BigscreenTheatre	/	60; 90; 120	*WebRTC*	15 Mbps
Tiešraides video straumēšanas simulācija	*Big_Buck_Bunny*	144 P–4 K	30; 60	*HLS*	*VBR*
	Big_Buck_Bunny	144 P–4 K	30; 60	*DASH*	*VBR*
	Big_Buck_Bunny	144 P–4 K	30; 60	*RTMP*	*VBR*
	Big_Buck_Bunny	144 P–4 K	30; 60	*HTTP_FLV*	*VBR*
	Big_Buck_Bunny	144 P–4 K	30; 60	*WebRTC*	*VBR*
	Big_Buck_Bunny	144 P–4 K	30; 60	*RTMP_Upload*	*VBR*

2.2. Klasifikācijas modeļi

Mūsdienās tīkla datu plūsmas klasifikācijas (*NTC*) uzdevumos tiek plaši lietoti mašīnmācīšanās un dziļās mācīšanās klasifikatori, kas demonstrē augstu un stabilu veiktspēju dažādām datu kopām [13]. 1., 2. un no 4. līdz 8. rakstam ir aprakstīts dažādu modeļu lietojums

– daudzslāņu perceptrons (*MLP*), konvolūcijas neironu tīkls (*CNN*), dziļais uzticības tīkls (*DBN*), *Extra Trees*, grafu neironu tīkls (*GNN*), grafu konvolūcijas tīkls (*GCN*) un *XGBoost* [147]–[149], [105], [57], [51], [49], [52]. Šos modeļus var iedalīt divās grupās: dziļās mācīšanās modeļi un ansambļa mācīšanās modeļi. Apskatot dziļās mācīšanās arhitektūras, *CNN* ir viens no visplašāk izmantotajiem modeļiem klasifikācijai. 7. un 8. rakstā aprakstīts *1D-CNN* modelis, ko veido trīs viendimensijas konvolūcijas slāņi, kuriem seko četri pilnībā savienoti slāņi, un katrā ir vairāk par 1000 neironiem. Pēdējais slānis izmanto *SoftMax* aktivācijas funkciju vairāku kategoriju klasifikācijai. Šīs arhitektūras svarīgais aspekts ir pakāpeniski samazinātais kodolu izmērs (5, 3 un 1) konvolūcijas slāņos, kas veicina smalku un precīzu lokālo modeļu izdalīšanu datu secībā. Šī konstrukcija ir īpaši efektīva laika rindu un plūsmu līmeņa klasifikācijas uzdevumos. Atbilstošā formula dota 1. vienādojumā.

$$y_j[t] = \sum_{i=1}^{C_{\text{input}}} \sum_{k=0}^{K-1} x_i[t \times S + k - P] \times W_{j,i,k} + b_j, \quad (1)$$

kur $y_j[t]$ – izvade laika momentā t izvades kanālam ar numuru j; $x_i[\cdot]$ – ievade no kanāla ar numuru i; $W_{j,i,k}$ – svara koeficients savienojumam starp ievades kanālu i un izvades kanālu j kodola indeksā k; b_j – nobīdes (*bias*) sastāvdaļa izvades kanālam j; C_{input} – ievades kanālu skaits; K – kodola izmērs; S – solis (*stride*); P – papildināto nolašu skaits (*padding*).

Šī arhitektūra tika tālāk attīstīta līdz *2D-CNN* variantam, aizvietojot vienas dimensijas konvolūcijas slāņus ar divdimensiju ekvivalentiem, bet saglabājot kopējo struktūru. Šī adaptācija ļauj modelim izmantot telpiskās funkciju hierarhijas ar lokālajiem uztverošajiem laukumiem, nodrošinot tādas priekšrocības kā parametru efektivitāte, translācijas invariance un svara koeficientu koplietošana. *2D-CNN* ir īpaši piemērots vērtuma līmeņa grafu klasifikācijai, kad ievades dati ir formatēti telpiskajā vai matriču formā. Atbilstošā konvolūcijas operācija ir definēta 2. vienādojumā un konceptuāli ir līdzīga *1D-CNN*.

$$y_j(h,w) = \sum_{i=1}^{C_{\text{input}}} \sum_{m=0}^{K_h-1} \sum_{n=0}^{K_w-1} x_i\left[h \times S_h + m - P_h, w \times S_w + n - P_w\right] \times W_{j,i,m,n} + b_j, \quad (2)$$

kur $y_j(h,w)$ – izvades vērtība telpiskajā pozīcijā (h,w) izvades kanālā ar numuru j; $x_i[\cdot\cdot]$ – ievades funkciju karte no kanāla ar numuru i un telpiskajām dimensijām (h,w); $W_{j,i,m,n}$ – kodola svars pozīcijā (m,n) ievades kanālam i un izvades kanālam j; K_h un K_w – konvolūcijas kodola dimensijas – augstums un platums; S_h un S_w– solis (*stride*) augstuma un platuma dimensijās; $P_h$un P_w – papildu nolašu skaits (*padding*) augstuma un platuma dimensijās; C_{input} – ievades kanālu skaits; b_j – nobīdes (*bias*) koeficients izvades kanālam j. *2D-CNN* modelis ir modificēts, balstoties *1D-CNN* modelī, kā redzams 2. tabulā.

Tabula 2.
2D-CNN modeļa arhitektūras apraksts

Slānis	**Tips**	**Kodola Izmērs**	**Filtri**	**Solis**	**Pildījums**	**Ievades Izmērs**	**Izvades Izmērs**
Ievades	–	–	–	–	–	(1, 100, 100)	(1, 100, 100)
Conv1	*Conv2D*	(5, 5)	32	1	2	(1, 100, 100)	(32, 100, 100)
Pool1	*MaxPool2D*	(2, 2)	–	2	0	(32, 100, 100)	(32, 50, 50)
Conv2	*Conv2D*	(3, 3)	64	1	1	(32, 50, 50)	(64, 50, 50)
Pool2	*MaxPool2D*	(2, 2)	–	2	0	(64, 50, 50)	(64, 25, 25)
Conv3	*Conv2D*	(1, 1)	128	1	0	(64, 25, 25)	(128, 25, 25)
Izlīdzināšana	Izlīdzināšanas (*Flatten*)	–	–	–	–	(128, 25, 25)	(80000)
FC1	Lineārs	–	1000	–	–	(80000)	(1000)
FC2	Lineārs	–	500	–	–	(1000)	(500)
FC3	Lineārs	–	250	–	–	(500)	(250)
FC4	Lineārs	–	125	–	–	(250)	(125)
Izvades	Lineārs	–	C	–	–	(125)	(C)

Papildus tika izstrādāts *RNN* modelis, kurā tika izmantoti trīs virknē apvienoti rekursīvie slāņi, lietojot *1D-CNN* arhitektūru. *RNN* (rekursīvie neironu tīkli) ir efektīvi laika rindu modelēšanā, jo tie spēj saglābāt nemainīgus iekšējos stāvokļus modeļa darbības laikā. Lai gan tie nav tik sarežģīti kā *LSTM* vai *Transformer* modeļi, *RNN* arhitektūra ir pietiekami efektīva īsterminā likumsakarību atklāšanai datu secībās. Atbilstošās apstrādes darbības apraksta 3. vienādojums.

$$h_t = \mathrm{Tanh}(W_{ih} \times x_t + b_{ih} + W_{hh} \times h_{t-1} + b_{hh}), \quad (3)$$

kur $h_t \in R^H$ – slēptais stāvoklis laika momentā t; $x_t \in R^D$ – ievades vektors laika momentā t; $W_{ih} \in R^{H \times D}$ – svaru matrica, kas savieno ievades un slēpto slāņus; $W_{hh} \in R^{H \times H}$ – svaru matrica no slēptā uz slēpto slāni; b_{ih} un $b_{hh} \in R^H$ – attiecīgie nobīdes vektori; $\mathrm{Tanh}(\cdot)$ – elementu hiperboliskā tangensa aktivācijas funkcija.

Plūsmas līmeņa klasifikācijas uzdevumā atsevišķās plūsmas līmeņa pazīmes ir atkarīgas no citām atvasinātām statistiskajām vai kontekstuālajām pazīmēm, tādējādi tās ir iespējams aprakstīt kā grafveida datus, kas ļauj izmantot *GCN* modeļus. Šādā lietojumā viena no datu kopas pazīmju dimensijām tiek izmantota, lai konstruētu grafu, kuram katra virsotne (mezgls) attēlo vienu paraugu, bet malas atspoguļo kaimiņu attiecības, izmantojot k-tuvāko kaimiņu (*k-NN*) algoritmu. Tas pārveido klasifikācijas uzdevumu par mezgla līmeņa problēmu. *GCN* modeļiem ir vairākas priekšrocības, piemēram, spēja mācīties no datiem, kas nav strukturēti pēc Eiklīda metrikas, apkopošana ar lokālajiem kaimiņu mezgliem un augstu precizitāti mezglu un grafu klasifikācijas uzdevumos. Matemātiskais formulējums redzams 4. vienādojumā.

$$H^{(l+1)} = \mathrm{ReLU}\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right), \quad (4)$$

kur $H^{(l)} \in R^{N \times F^{(l)}}$ – mezglu pazīmju matrica l līmenī; $W^{(l)} \in R^{F^{(l)} \times F^{(l+1)}}$ – apmācāmā svaru matrica; $\tilde{A} = A + I$ – blakusmatrica ar pašsaitēm (*self-loops*); $\tilde{D}$ – $\tilde{A}$ atbilstošā diagonālā pakāpes matrica; $ReLU(\cdot)$ – izlīdzināšanās aktivācijas funkcija (*rectifier activation function*).

Salīdzināšanai ar klasifikācijas veiktspēju tika izvēlēti trīs references modeļi: *Deep Packet, FlowPic* un *FS-Net* [150]–[152]. Gan *Deep Packet*, gan *FlowPic* pamatā ir *1D-CNN* arhitektūras, savukārt *FS-Net* izmanto *Gated Recurrent Unit* (*GRU*) – uzlabotu rekurento modeli, kas tika papildināts ar atjaunināšanas un atiestatīšanas elementiem (*update and reset gates*), lai labāk pārvaldītu informācijas plūsmu un atrisinātu gradienta izzušanas (*vanishing gradient*) problēmu. Lai vēl vairāk uzlabotu modeļu vispārināšanas spējas, tika lietotas ansambļa modeļu mācīšanās metodes. Slāņu modelis tiek veidots no 11 heterogēniem pamatmodeļiem, kuros tiek izmantotas dažādas pastiprināšanas (*boosting*) un modeļu saknēšanas apvienošanas (*bootstrap aggregating – bagging*) pieejas. Galīgais klasifikācijas rezultāts tiek ģenerēts, izmantojot loģistisko regresiju, tādējādi izmantojot individuālo klasifikatoru stiprās puses kopējās veiktspējas uzlabošanai, kā aprakstīts 1. publikācijā.

3. tabulā apkopota visu novērtēto klasifikācijas modeļu skaitļošanas sarežģītība. Slāņu modelim ir visaugstākā sarežģītība vairāku koka modeļu apvienošanas dēļ, kā arī papildu noslodzes dēļ, ko veido visu koka modeļu paralēlā apmācīšana. *2D-CNN* modelim ir salīdzinoši augstas skaitļošanas un atmiņas izmaksas uz vienu sēriju (*batch*), jo īpaši apstrādājot augstas izšķirtspējas ievades un izmantojot dziļos, pilnībā savienotos slāņus. Gan *FS-Net*, gan *RNN* modeļi darbojas divvirzienu režīmā ar secībām, tāpēc skaitļošanas sarežģītība pieaug līdz ar slāņu skaitu, padarot tos salīdzinoši prasīgus. *1D-CNN* modelim ir augsta skaitļošanas sarežģītība, jo tas izmanto vairāk nekā 1000 neironus pirmajā pilnībā savienotajā slānī. Savukārt *GCN* modeļa skaitļošanas sarežģītība galvenokārt ir atkarīga no grafa malu skaita. Tā kā katram grafam tiek izmantoti tikai trīs tuvākie kaimiņi, sarežģītība saglabājas salīdzinoši zema. No visiem modeļiem *Deep Packet* un *FlowPic* demonstrē viszemāko skaitļošanas sarežģītību, pateicoties to vienkāršajai *1D-CNN* arhitektūrai un efektīvajiem pazīmju iegūšanas mehānismiem.

Tabula 3.
Aprēķinu sarežģītības kopsavilkums visiem klasifikācijas modeļiem

Modeļi	**Skaitļošanas sarežģītība**
1D-CNN [52]	$\mathcal{O}(B \times L \times F_{1D})$
RNN	$\mathcal{O}\left(B \times L \times n_{\text{layers}} \times (D \times H + H^2)\right)$
GCN [57]	$\mathcal{O}\left(B \times \left(N \times {F_{gcn}}^2 + \lvert E \rvert \times F_{gcn}\right)\right)$
Stacking (slāņu)	$\mathcal{O}(M \times T \times N \times \log N) + \mathcal{O}\big((M + D) \times N\big)$
2D-CNN	$\mathcal{O}(B \times H_s \times W \times F_{2D})$
Deep Packet [150]	$\mathcal{O}(B \times D \times H^2)$
FlowPic [151]	$\mathcal{O}(B \times D \times H^2)$
FS-Net [152]	$\mathcal{O}\big(B \times L \times H^2 \times n_{\text{layers}}\big)$

Papildu piezīmes:
B – sērijas izmērs (*batch size*);
C – klašu skaits (*number of classes*);
D – ieejas pazīmju vektora garums (*input feature vector length*);
F – izvades kanālu skaits (*number of output channels*);
F_{1D} – kopējais filtru skaits *1D* konvolūcijas slāņos (*total filter operations in 1D convolutional layers*);
F_{2D} – kopējais filtru skaits *2D* konvolūcijas slāņos (*total filter operations in 2D convolutional layers*);
L – *1D* secības garums *1D-CNN/RNN* modeļos (*length of 1D sequence in 1D-CNN/RNN*);
n_{layers} – *RNN/GRU* slāņu skaits (*number of RNN/GRU layers*);
F_{gcn} – katra mezgla pazīmju dimensija (*feature dimension per node*);
H – slēpto vienību skaits *RNN/GRU* slāņos (*number of hidden units in RNN/GRU layers*);
H_s – *2D* ieejas telpiskais augstums (*spatial height of 2D input*);
W – *2D* ieejas telpiskais platums (*spatial width of 2D input*);
$|E|$ – malu skaits grafā (*number of edges*);
N – apmācāmo paraugu skaits (*number of training samples*);
M – meta modeļu skaits (*number of meta learners*);
T – novērtētāju skaits (*number of estimators*).

3. GALVENIE REZULTĀTI

3.1. Laika rindas līmeņa klasifikācijas veiktspēja

Interaktīvo mediju lietotņu datu plūsmas klasifikācijas veiktspējas novērtējumā laika rindas līmenī tika izmantoti gan pakešu, gan baitu secību dati par laiku. Kopumā tika lietoti septiņi klasifikācijas modeļi, veicot 28 klasifikācijas eksperimentus katrā datu kopā – tiešraides straumēšana, mākoņspēles un tiešraides video straumēšanas simulācija. Metavides datu kopa laika rindas līmeņa klasifikācijai netika iekļauta eksperimentos, jo īsais neapstrādātās datu plūsmas ilgums neļāva izveidot pietiekami garas laika rindas. 4. tabulā dots apkopojums, saskaņā ar kuru redzams, ka *1D-CNN* modelis sasniedza visaugstāko precizitāti tiešraides straumēšanas datu kopā – 0,65 gan pakešu, gan baitu laika rindās. Neveicot *SMOTEENN* mazāk pārstāvētās klases paraugu paplašināšanu, slāņu modeļa rezultāti bija par 15 % zemāki, salīdzinot ar *1D-CNN* modeļa rezultātiem. Lietojot *SMOTEENN* paraugu paplašināšanu, *1D-CNN* modeļa klasifikācijas precizitāte uzlabojās par vairāk nekā 30 %, sasniedzot 0,95 pakešu laika rindās ar 95 kategorijām. Līdzīgi uzlabojumi bija novērojami visiem modeļiem arī baitu laika rindās.

Īpaši jāuzsver, ka pēc *SMOTEENN* paraugu paplašināšanas veiktspējas atšķirība starp slāņu un *1D-CNN* modeļiem būtiski samazinājās – slāņu modelis sasniedza klasifikācijas precizitāti, kas gan pakešu, gan baitu laika rindas datos bija par 5 % zemāka nekā *1D-CNN* modelim. Ņemot vērā lielo kategoriju skaitu un neizlīdzinātu (nesabalancēto) paraugu sadalījumu, par piemērotāku tika izvēlēts makro F1 rādītājs, kas vienlīdzīgi vērtē visas kategorijas, neizceļot

tās, kas ietver lielu paraugu apjomu. Augstāko makro F1 rādītāju (0,93) sasniedza *1D-CNN* modelis pakešu laika rindās pēc *SMOTEENN* paraugu papildināšanas. 5. tabulā apkopoti detalizēti rezultāti pa kategorijām, kur lielākā daļa kategoriju uzrādīja F1 vērtības virs 0,8. Tomēr dažām kategorijām bija zema precizitāte un atbilstības rādītājs (*recall*). Piemēram, kategorija *480P_3Mbps_N3_10MHz*, neskatoties uz lielo paraugu skaitu, uzrādīja tikai 0,68 atbilstības rādītāju, daudzi paraugi tika nepareizi klasificēti kā *1080P_60FPS_VLS_YouTube*. Šīm kategorijām ir zema datu plūsmas līdzība – *480P_3Mbps_N3_10MHz* vidēji ap 50 pakešu uz 100 ms, bet *1080P_60FPS_VLS_YouTube* ap 200 pakešu uz 100 ms. Zemo veiktspēju, visticamāk, veicināja *SMOTE* ģenerētie interpolētie paraugi trokšņainās vai pārklājošās zonās, neņemot vērā starpklašu distances, kas radīja kļūdainas klasifikācijas. Tāpat kategorijas *1440P_60FPS_VLS_BiliBili* un *1080P_60FPS_VoD_YouTube* bieži tika sajauktas ar *1080P_60FPS_VLS_YouTube*, jo tām ir ļoti līdzīgas datu plūsmas – vidēji ap 300 paketēm uz 100 ms. Tāpēc zema klasifikācijas veiktspēja nav tikai neliela paraugu apjoma sekas, bet arī paraugu papildināšana (*resampling*) procesa trokšņa pastiprināšanās apstākļos būtiski ietekmē rezultātus.

Tiešraides straumēšanas datu kopā pēc *SMOTEENN* lietošanas kopējais paraugu skaits palielinājās par vairāk nekā 25 %. Taču sākotnēji nelielajā un kategoriju ziņā sabalansētajā mākoņspēļu datu kopā paraugu skaits samazinājās vairāk nekā par 30 %, un katrā kategorijā bija mazāk par 50 paraugiem. Šādu samazinājumu izraisīja *SMOTEENN* metodes *ENN* komponente, kas agresīvi dzēš paraugus, it īpaši sabalansētus, kuriem ir stipri trokšņi vai arī tiek pārklātās to kategorijas. Lai gan augstākā klasifikācijas precizitāte, kas sasniegta pakešu laika rindām, bija 0,94, izmantojot *1D-CNN* modeli, ierobežotais paraugu skaits samazina šī rezultāta ticamību. Kā redzams 7. tabulā, trīs kategorijās – *SP_720P_VP9_WebRTC*, *TR_2160P_VP9_WebRTC* un *TR_1080P_VP9_Play_State* – netika veikta neviena pareiza klasifikācija. Tas ir izskaidrojams ar to, ka pārbaudes datu kopā šīm kategorijām nebija atbilstošu paraugu, un tas būtiski samazināja gan precizitāti, gan atbilstības rādītāju. Pārskatot izejas datu failus, paliek skaidrs, ka, lai gan datu ilgums pārsniedza 600 sekundes, tas joprojām nebija pietiekams, lai izveidotu kvalitatīvu un labi sabalansētu laika rindas līmeņa datu kopu.

Tiešraides video straumēšanas simulācijas datu kopā izejas datu plūsmas ilgums arī bija aptuveni 600 s. Tomēr, ievērojot kopējo paraugu skaitu, kas pārsniedza 10 000 nolases, lielā datu kopa ļāva *SMOTE* ģenerēt uzticamākus sintētiskos paraugus. Turklāt *ENN* guva labumu no blīvas datu struktūras, kas samazināja kļūdainas klasifikācijas varbūtību. Lietojot *SMOTEENN*, visas kategorijas tika pareizi papildinātas, un katras kategorijas paraugu apjoms vairāk nekā divkāršojies. Saskaņā ar 8. tabulā sniegtajiem klasifikācijas rezultātiem bez paraugu papildināšanas visi modeļi sākotnēji sasniedza klasifikācijas precizitāti zem 0,5 gan pakešu, gan baitu laika rindās. Tas liecina par precīzas klasifikācijas grūtībām, ja paraugu skaits nav pietiekams. Lietojot *SMOTEENN*, labāku rezultātu baitu laika rindās sasniedza *1D-CNN* modelis ar precizitāti (0,91) 78 kategorijās un makro F1 rādītāju (0,90). Salīdzinot ar pakešu laika rindām, šis rezultāts parāda vairāk nekā 7 % klasifikācijas veiktspējas pieaugumu identiskos apstākļos. Šo uzlabojumu var izskaidrot ar to, ka dažas kategorijas, piemēram, *WebRTC*, ģenerē lielu pakešu apjomu, kas palielina datu troksni un padara klasifikāciju pakešu līmenī sarežģītāku. Slāņu modelis arī darbojās labi, sasniedzot precizitāti virs 0,8, lai gan par

aptuveni 8 % zemāku nekā *1D-CNN* modelim. Detalizētā analīze 11. tabulā parāda, ka kategorijai *360P30FPS_HLS_VLS_Download* precizitāte bija tikai 0,47, galvenokārt biežo kļūdu dēļ, kurās tika veikta klasifikācija kā *240P30FPS_HLS_VLS_Download* kategorija ar atbilstības rādītāju 0,59. Kategorijas *480P30FPS_HLS_VLS_Download* un *720P60FPS_RTMP_VLS_Upload* bieži tika sajauktas, kas negatīvi ietekmēja galaprecizitāti. Kategorijas *2160P30FPS_HLS_VLS_Download* un *2160P60FPS_HLS_VLS_Download* arī bija grūti atšķiramas to savstarpējās līdzības dēļ.

Kopumā gan *1D-CNN*, gan slāņu modelis demonstrēja augstu klasifikācijas veiktspēju laika rindas līmenī, kur vislabākos un visstabilākos rezultātus uzrādīja *1D-CNN* modelis. Lielākā daļa kategoriju sasniedza augstu precizitāti un atbilstības rādītāju, lai gan dažām kategorijām zemāka precizitāte bija saistīta gan ar modeļu ierobežojumiem, gan ar pārmērīgu datu troksni. Tomēr jāņem vērā arī tas, ka šī augstā klasifikācijas veiktspēja bija būtiski atkarīga no paraugu papildināšanas tehnikām. Šo tehniku efektivitāti ievērojami ietekmē vairāki būtiski faktori – datu kopas apjoms, pazīmju variācija, klases nelīdzsvarotība un to atšķiršanas spēja. Lielākas datu kopas nodrošina blīvāku un precīzāk pārstāvētu pazīmju telpu, kas ļauj *SMOTE* ģenerēt nozīmīgākus un daudzveidīgākus sintētiskos paraugus. Augsta pazīmju variācija ļauj veikt interpolāciju starp atšķirīgiem paraugiem, samazinot neskaidru vai trokšņainu punktu iespējamību. Gadījumos, kad pastāv patiesa klašu nelīdzsvarotība, *SMOTE* efektīvi uzlabo minoritātes klašu paraugu apjomu – to pārstāvniecību. Tomēr līdzsvarotās vai zemas variācijas datu kopās *SMOTE* var ģenerēt zemas kvalitātes vai liekus paraugus. Ja klašu atšķiršanas spēja ir zema, sintētiskie paraugi var šķērsot lēmumu robežas, it īpaši kombinācijā ar agresīvām trokšņa dzēšanas metodēm (piemērām, *ENN*), kas var pasliktināt datu kvalitāti, dzēšot citādi derīgos, bet savstarpēji pārklājošos paraugus. Tāpēc sintētisko paraugu metožu iznākumi ir ļoti atkarīgi no datu īpašībām. Ja uzmanība tiek pievērsta tikai reāliem, nepapildinātajiem (nesintētiskajiem) paraugiem, augstas kvalitātes datu ieguves nozīme ievērojami pieaug. Aktīva tīkla datu plūsmas uztveršana ilgākā laika posmā (vairāk nekā 30 minūtes) var palielināt datu vākšanas sarežģītību, taču tā ir nepieciešama, lai nodrošinātu pietiekamu datu apjomu. Ir būtiski ņemt vērā pakešu vai baitu līmeņa datu kvalitāti, piemēram, zemas izšķirtspējas un zemā kadru ātruma avoti var radīt mazo pakešu lielu skaitu, kuriem būs zemā noslodze, rezultātā palielinot trokšņus un apgrūtinot klasifikācijas procesu. Šādus trokšņus ir rūpīgi jānotīra, lai uzlabotu modeļu veiktspēju.

Tabula 4.

Klasifikācijas rezultātu kopsavilkums tiešraides straumēšanas datu kopām laika rindas līmenī

Laika rinda	T_{smp}	T_{wnd}	**Diskretizācija**	**Modelis**	**Kategorija**	***P.W***	***R.W***	***F.A***	***F.I***	***A.Test***	***A.Train***	***A.Validate***
Packet	100 ms	5 s	/	*1D-CNN*	95	0,72	0,65	0,66	0,65	0,65	0,75	0,75
Packet	100 ms	5 s	/	*RNN*	95	0,43	0,42	0,38	0,42	0,41	0,42	0,43
Packet	100 ms	5 s	/	*GCN*	95	0,49	0,43	0,43	0,43	0,43	0,50	0,42
Packet	100 ms	5 s	/	*Stacking*	95	0,60	0,58	0,59	0,58	0,58	0,58	0,58
Packet	100 ms	5 s	/	*Deep Packet*	95	0,57	0,54	0,54	0,54	0,55	0,58	0,57
Packet	100 ms	5 s	/	*FlowPic*	95	0,47	0,46	0,43	0,46	0,46	0,47	0,48
Packet	100 ms	5 s	/	*FS-Net*	95	0,52	0,51	0,50	0,51	0,51	0,54	0,53

4. tabulas turpinājums

Byte	100 ms	5 s	/	*1D-CNN*	95	0,72	0,65	0,66	0,65	0,65	0,77	0,76
Byte	100 ms	5 s	/	*RNN*	95	0,43	0,42	0,39	0,42	0,42	0,42	0,42
Byte	100 ms	5 s	/	*GCN*	95	0,46	0,39	0,41	0,39	0,39	0,52	0,42
Byte	100 ms	5 s	/	*Stacking*	95	0,65	0,63	0,63	0,63	0,63	0,82	0,62
Byte	100 ms	5 s	/	*Deep Packet*	95	0,57	0,55	0,55	0,55	0,55	0,59	0,58
Byte	100 ms	5 s	/	*FlowPic*	95	0,47	0,47	0,44	0,47	0,46	0,48	0,46
Byte	100 ms	5 s	/	*FS-Net*	95	0,55	0,51	0,50	0,51	0,50	0,54	0,54
Packet	100 ms	5 s	*SMOTEENN*	*1D-CNN*	95	0,96	0,95	0,93	0,95	0,95	0,98	0,97
Packet	100 ms	5 s	*SMOTEENN*	*RNN*	95	0,75	0,75	0,66	0,75	0,75	0,76	0,76
Packet	100 ms	5 s	*SMOTEENN*	*GCN*	95	0,84	0,82	0,75	0,82	0,82	0,82	0,79
Packet	100 ms	5 s	*SMOTEENN*	*Stacking*	95	0,90	0,90	0,82	0,90	0,90	0,99	0,90
Packet	100 ms	5 s	*SMOTEENN*	*Deep Packet*	95	0,83	0,83	0,75	0,83	0,82	0,84	0,84
Packet	100 ms	5 s	*SMOTEENN*	*FlowPic*	95	0,73	0,73	0,64	0,73	0,74	0,74	0,74
Packet	100 ms	5 s	*SMOTEENN*	*FS-Net*	95	0,83	0,82	0,76	0,82	0,82	0,86	0,86
Byte	100 ms	5 s	*SMOTEENN*	*1D-CNN*	95	0,96	0,94	0,91	0,94	0,94	0,97	0,97
Byte	100 ms	5 s	*SMOTEENN*	*RNN*	95	0,73	0,73	0,66	0,73	0,73	0,75	0,77
Byte	100 ms	5 s	*SMOTEENN*	*GCN*	95	0,83	0,81	0,76	0,81	0,81	0,82	0,78
Byte	100 ms	5 s	*SMOTEENN*	*Stacking*	95	0,91	0,91	0,84	0,91	0,91	0,99	0,91
Byte	100 ms	5 s	*SMOTEENN*	*Deep Packet*	95	0,83	0,82	0,76	0,82	0,82	0,84	0,84
Byte	100 ms	5 s	*SMOTEENN*	*FlowPic*	95	0,72	0,72	0,63	0,72	0,72	0,72	0,73
Byte	100 ms	5 s	*SMOTEENN*	*FS-Net*	95	0,86	0,85	0,79	0,85	0,85	0,87	0,87

Tabula 5.

Klasifikācijas veiktspēja katrā tiešraides straumēšanas datu kopas kategorijā, izmantojot 1D-CNN modeli pēc paraugu papildināšanas (resampling) pakešu laika rindas līmenī

Kategorija	***P.***	***R.***	***F.***	**Kategorija**	***P.***	***R.***	***F.***
480P_3Mbps_N3_10Mhz	0,98	0,68	0,8	*VPN_Vimeo*	1	1	1
480P_3Mbps_N78_10Mhz	0,96	0,91	0,93	*VPN_YouTube*	0,98	0,78	0,87
480P_3Mbps_N78_20Mhz	0,94	0,92	0,93	*160P_30FPS_VoD_Twitch*	0,99	0,91	0,95
720P_6Mbps_N3_10Mhz	0,99	0,96	0,97	*160P_30FPS_VLS_Twitch*	1	0,99	1
720P_6Mbps_N78_10Mhz	0,99	1	1	*360P_30FPS_VoD_Twitch*	1	0,9	0,95
720P_6Mbps_N78_20Mhz	0,96	0,98	0,97	*360P_30FPS_VLS_Twitch*	0,97	0,99	0,98
1080P_10Mbps_N3_10Mhz	0,99	1	1	*480P_30FPS_VoD_Twitch*	1	1	1
1080P_10Mbps_N78_10Mhz	0,98	1	0,99	*480P_30FPS_VLS_Twitch*	0,97	0,98	0,97
1080P_10Mbps_N78_20Mhz	0,97	0,93	0,95	*720P_30FPS_VoD_Twitch*	1	1	1
2160P_20Mbps_N3_10Mhz	0,99	0,96	0,98	*720P_60FPS_VoD_Twitch*	1	1	1
2160P_20Mbps_N78_20Mhz	1	0,99	0,99	*720P_30FPS_VLS_Twitch*	0,95	0,89	0,92
4320P_40Mbps_N3_10Mhz	0,95	0,97	0,96	*720P_60FPS_VLS_Twitch*	0,96	0,94	0,95
4320P_40Mbps_N78_20Mhz	0,97	0,99	0,98	*1080P_30FPS_VoD_Twitch*	0,97	0,96	0,97
2160P_30FPS_VoD_BiliBili	0,99	1	1	*1080P_60FPS_VoD_Twitch*	0,97	0,98	0,97
2160P_60FPS_VoD_BiliBili	1	0,95	0,98	*1080P_30FPS_VLS_Twitch*	0,99	1	1
2160P_120FPS_VoD_BiliBili	0,99	1	1	*1080P_60FPS_VLS_Twitch*	0,95	0,95	0,95
2160P_60FPS_VLS_BiliBili	0,99	0,99	0,99	*240P_30FPS_VoD_Vimeo*	0,98	0,94	0,96
2160P_80FPS_VLS_BiliBili	0,83	1	0,91	*360P_30FPS_VoD_Vimeo*	1	0,94	0,97
4320P_30FPS_VoD_BiliBili	1	0.93	0,96	*540P_30FPS_VoD_Vimeo*	0,77	0,89	0,83
360P_30FPS_VoD_BiliBili	0,93	0,97	0,95	*720P_30FPS_VoD_Vimeo*	0,95	0,97	0,96
480P_30FPS_VoD_BiliBili	0,98	0,97	0,98	*1080P_30FPS_VoD_Vimeo*	0,82	0,87	0,84
480P_30FPS_VLS_BiliBili	0,98	0,98	0,98	*1440P_30FPS_VoD_YouTube*	0,96	0,97	0,96
720P_30FPS_VoD_BiliBili	0,97	1	0,99	*1440P_60FPS_VoD_YouTube*	0,91	0,82	0,86
720P_30FPS_VLS_BiliBili	0,95	0,92	094	*1440P_30FPS_VLS_YouTube*	0,98	1	0,99
1080P_30FPS_VoD_BiliBili	1	0,9	0,95	*1440P_60FPS_VLS_YouTube*	0,93	0,97	0,95
1080P_60FPS_VoD_BiliBili	0,96	0,96	0,96	*2160P_30FPS_VoD_YouTube*	0,9	0,91	0,9
1080P_60FPS_VLS_BiliBili	0,64	0,78	0,7	*2160P_60FPS_VoD_YouTube*	0,96	1	0,98
1440P_60FPS_VLS_BiliBili	0,75	0,75	0,75	*2160P_30FPS_VLS_YouTube*	0,99	1	0,99
144P_30FPS_VOD_Facebook	0,87	0,85	0,86	*2160P_60FPS_VLS_YouTube*	1	0,98	0,99
144P_30FPS_VLS_Facebook	0,8	0,8	0,8	*4320P_30FPS_VoD_YouTube*	0,99	0,89	0,93
240P_30FPS_VLS_Facebook	0,98	0,96	0,97	*4320P_60FPS_VoD_YouTube*	0,99	1	1
360P_30FPS_VOD_Facebook	0,79	0,9	0,84	*144P_30FPS_VoD_YouTube*	0,97	1	0,98

5. tabulas turpinājums

360P_30FPS_VLS_Facebook	0,92	0,63	0,75	*144P_30FPS_VLS_YouTube*	0,96	0,95	0,95
480P_30FPS_VOD_Facebook	0,97	0,99	0,98	*240P_30FPS_VoD_YouTube*	0,98	0,97	0,98
480P_30FPS_VLS_Facebook	1	0,8	0,89	*240P_30FPS_VLS_YouTube*	1	1	1
540P_30FPS_VOD_Facebook	0,92	0,8	0,86	*360P_30FPS_VoD_YouTube*	1	0,99	1
720P_30FPS_VOD_Facebook	0,95	0,97	0,96	*360P_30FPS_VLS_YouTube*	0,97	0,97	0,97
720P_30FPS_VLS_Facebook	0,88	0,95	0,91	*480P_30FPS_VoD_YouTube*	0,98	0,93	0,96
1080P_30FPS_VOD_Facebook	0,98	0,93	0,96	*480P_30FPS_VLS_YouTube*	0,92	1	0,96
1080P_30FPS_VLS_Facebook	0,87	0,99	0,93	*720P_30FPS_VoD_YouTube*	0,89	0,73	0,8
360P_30FPS_VLS_TikTok	1	1	1	*720P_60FPS_VoD_YouTube*	0,95	0,91	0,93
540P_30FPS_VLS_TikTok	1	0,99	1	*720P_30FPS_VLS_YouTube*	0,94	0,96	0,95
720P_30FPS_VLS_TikTok	0,97	0,98	0,98	*720P_60FPS_VLS_YouTube*	0,99	0,99	0,99
720P_60FPS_VLS_TikTok	0,99	1	1	*1080P_30FPS_VoD_YouTube*	0,77	0,83	0,8
1080P_30FPS_VLS_TikTok	1	1	1	*1080P_60FPS_VoD_YouTube*	1	0,11	0,2
1080P_60FPS_VLS_TikTok	1	0,99	1	*1080P_30FPS_VLS_YouTube*	0,97	0,99	0,98
Tor_Vimeo	0,95	0,98	0,97	*1080P_60FPS_VLS_YouTube*	0,38	0,94	0,54
Tor_YouTube	0,97	0,95	0,96				

Tabula 6.

Klasifikācijas veiktspējas kopsavilkums mākoņspēļu datu kopām laika rindas līmenī

Laika rinda	T_{smp}	T_{wnd}	**Diskretizācija**	**Modelis**	**Kategorija**	***P.W***	***R. W***	***F. A***	***F. I***	***A. Test***	***A. Train***	***A. Validate***
Packet	100 ms	5 s	/	*1D-CNN*	12	0,58	0,63	0,55	0,63	0,63	0,61	0,59
Packet	100 ms	5 s	/	*RNN*	12	0,49	0,53	0,47	0,52	0,50	0,51	0,51
Packet	100 ms	5 s	/	*GCN*	12	0,30	0,38	0,29	0,38	0,38	0,46	0,37
Packet	100 ms	5 s	/	*Stacking*	12	0,65	0,64	0,64	0,64	0,64	0,98	0,61
Packet	100 ms	5 s	/	*Deep Packet*	12	0,49	0,49	0,45	0,49	0,46	0,49	0,47
Packet	100 ms	5 s	/	*FlowPic*	12	0,45	0,47	0,41	0,47	0,43	0,43	0,44
Packet	100 ms	5 s	/	*FS-Net*	12	0,56	0,53	0,51	0,52	0,52	0,51	0,51
Byte	100 ms	5 s	/	*1D-CNN*	12	0,57	0,65	0,55	0,58	0,52	0,53	0,52
Byte	100 ms	5 s	/	*RNN*	12	0,56	0,58	0,52	0,53	0,47	0,51	0,48
Byte	100 ms	5 s	/	*GCN*	12	0,38	0,37	0,31	0,37	0,37	0,48	0,39
Byte	100 ms	5 s	/	*Stacking*	12	0,50	0,48	0,48	0,48	0,48	0,98	0,48
Byte	100 ms	5 s	/	*Deep Packet*	12	0,43	0,45	0,41	0,45	0,48	0,48	0,48
Byte	100 ms	5 s	/	*FlowPic*	12	0,41	0,43	0,38	0,43	0,42	0,45	0,42
Byte	100 ms	5 s	/	*FS-Net*	12	0,57	0,59	0,51	0,53	0,48	0,49	0,48
Packet	100 ms	5 s	*SMOTEENN*	*1D-CNN*	12	0,95	0,97	0,95	0,96	0,94	0,96	0,89
Packet	100 ms	5 s	*SMOTEENN*	*RNN*	12	0,76	0,77	0,65	0,74	0,69	0,76	0,65
Packet	100 ms	5 s	*SMOTEENN*	*GCN*	12	0,54	0,64	0,33	0,64	0,64	0,76	0,64
Packet	100 ms	5 s	*SMOTEENN*	*Stacking*	12	0,74	0,74	0,72	0,73	0,73	0,97	0,73
Packet	100 ms	5 s	*SMOTEENN*	*Deep Packet*	12	0,79	0,81	0,71	0,79	0,73	0,76	0,74
Packet	100 ms	5 s	*SMOTEENN*	*FlowPic*	12	0,64	0,70	0,59	0,64	0,59	0,61	0,51
Packet	100 ms	5 s	*SMOTEENN*	*FS-Net*	12	0,82	0,88	0,80	0,79	0,74	0,75	0,79
Byte	100 ms	5 s	*SMOTEENN*	*1D-CNN*	12	0,93	0,90	0,89	0,89	0,88	0,86	0,96
Byte	100 ms	5 s	*SMOTEENN*	*RNN*	12	0,87	0,87	0,82	0,85	0,82	0,79	0,79
Byte	100 ms	5 s	*SMOTEENN*	*GCN*	12	0,68	0,65	0,42	0,65	0,65	0,81	0,71
Byte	100 ms	5 s	*SMOTEENN*	*Stacking*	12	0,63	0,63	0,59	0,63	0,62	0,97	0,61
Byte	100 ms	5 s	*SMOTEENN*	*Deep Packet*	12	0,78	0,79	0,67	0,78	0,82	0,78	0,83
Byte	100 ms	5 s	*SMOTEENN*	*FlowPic*	12	0,73	0,79	0,67	0,76	0,69	0,66	0,74
Byte	100 ms	5 s	*SMOTEENN*	*FS-Net*	12	0,84	0,80	0,72	0,79	0,78	0,77	0,79

Tabula 7.
Klasifikācijas veiktspēja katrā makoņspēļu datu kopas kategorijā, izmantojot 1D-CNN modeli pēc paraugu papildināšanas (resampling) pakešu laika rindas līmenī

Kategorija	*P.*	*R.*	*F.*
SP_2160P_VP9_WebRTC	0,67	0,67	0,67
SP_720P_VP9_WebRTC	0	0	0
SP_1080P_H264_WebRTC	1	1	1
SP_1080P_VP9_WebRTC	0,88	1	0,94
SP_1080P_VP9_RTP	1	1	1
SP_1080P_VP9_Play_State	0,89	1	0,94
TH_1080P_VP9_RTP	1	1	1
TR_2160P_VP9_WebRTC	0	0	0
TR_720P_VP9_WebRTC	1	0,92	0,96
TR_1080P_VP9_WebRTC	1	1	1
TR_1080P_VP9_RTP	1	1	1
TR_1080P_VP9_ Play_State	0	0	0

Tabula 8.
Klasifikācijas veiktspējas kopsavilkums tiešraides video straumēšanas simulācijas datu kopām laika rindas līmenī

Laika rinda	T_{smp}	T_{wnd}	Diskretizācija	Modelis	Kategorija	*P. W*	*R.W*	*F. A*	*F. I*	*A. Test*	*A. Train*	*A. Validate*
Packet	100 ms	5 s	/	*1D-CNN*	78	0,45	0,44	0,39	0,44	0,43	0,56	0,55
Packet	100 ms	5 s	/	*RNN*	78	0,30	0,33	0,26	0,31	0,30	0,34	0,34
Packet	100 ms	5 s	/	*GCN*	78	0,34	0,35	0,31	0,35	0,35	0,51	0,35
Packet	100 ms	5 s	/	*Stacking*	78	0,34	0,34	0,30	0,34	0,34	0,91	0,34
Packet	100 ms	5 s	/	*Deep Packet*	78	0,34	0,34	0,30	0,34	0,35	0,40	0,38
Packet	100 ms	5 s	/	*FlowPic*	78	0,29	0,31	0,27	0,31	0,29	0,33	0,34
Packet	100 ms	5 s	/	*FS-Net*	78	0,33	0,33	0,29	0,33	0,34	0,39	0,39
Byte	100 ms	5 s	/	*1D-CNN*	78	0,45	0,45	0,41	0,45	0,45	0,69	0,67
Byte	100 ms	5 s	/	*RNN*	78	0,28	0,28	0,23	0,28	0,27	0,31	0,30
Byte	100 ms	5 s	/	*GCN*	78	0,27	0,25	0,22	0,25	0,25	0,46	0,29
Byte	100 ms	5 s	/	*Stacking*	78	0,33	0,33	0,29	0,33	0,33	0,93	0,32
Byte	100 ms	5 s	/	*Deep Packet*	78	0,34	0,34	0,31	0,34	0,32	0,40	0,39
Byte	100 ms	5 s	/	*FlowPic*	78	0,26	0,26	0,22	0,26	0,27	0,30	0,30
Byte	100 ms	5 s	/	*FS-Net*	78	0,30	0,31	0,25	0,30	0,30	0,34	0,34
Packet	100 ms	5 s	*SMOTEENN*	*1D-CNN*	78	0,85	0,84	0,83	0,84	0,84	0,89	0,91
Packet	100 ms	5 s	*SMOTEENN*	*RNN*	78	0,41	0,42	0,37	0,41	0,40	0,41	0,40
Packet	100 ms	5 s	*SMOTEENN*	*GCN*	78	0,77	0,76	0,74	0,76	0,76	0,78	0,70
Packet	100 ms	5 s	*SMOTEENN*	*Stacking*	78	0,83	0,83	0,82	0,83	0,83	0,99	0,84
Packet	100 ms	5 s	*SMOTEENN*	*Deep Packet*	78	0,56	0,57	0,54	0,57	0,57	0,60	0,59
Packet	100 ms	5 s	*SMOTEENN*	*FlowPic*	78	0,44	0,45	0,41	0,45	0,45	0,45	0,45
Packet	100 ms	5 s	*SMOTEENN*	*FS-Net*	78	0,55	0,54	0,51	0,54	0,55	0,56	0,57
Byte	100 ms	5 s	*SMOTEENN*	*1D-CNN*	78	0,91	0,91	0,90	0,91	0,91	0,96	0,96
Byte	100 ms	5 s	*SMOTEENN*	*RNN*	78	0,42	0,42	0,41	0,42	0,42	0,45	0,43
Byte	100 ms	5 s	*SMOTEENN*	*GCN*	78	0,78	0,76	0,76	0,76	0,76	0,80	0,70
Byte	100 ms	5 s	*SMOTEENN*	*Stacking*	78	0,82	0,82	0,79	0,82	0,82	0,99	0,82
Byte	100 ms	5 s	*SMOTEENN*	*Deep Packet*	78	0,62	0,61	0,59	0,61	0,60	0,65	0,65
Byte	100 ms	5 s	*SMOTEENN*	*FlowPic*	78	0,43	0,44	0,42	0,44	0,45	0,47	0,46
Byte	100 ms	5 s	*SMOTEENN*	*FS-Net*	78	0,66	0,65	0,64	0,65	0,66	0,70	0,72

Tabula 9.

Klasifikācijas veiktspēja katrā tiešraides video simulācijas datu kopas kategorijā, izmantojot 1D-CNN modeli pēc pārmērošanas baitu laika rindas līmenī

Kategorija	*P.*	*R.*	*F.*	Kategorija	*P.*	*R.*	*F.*
144P30FPS_DASH_VLS_Download	0,8	0,74	0,77	*720P60FPS_RTMP_VLS_Download*	0,93	0,93	0,93
144P30FPS_HLS_VLS_Download	0,97	1	0,98	*720P60FPS_RTMP_VLS_Upload*	1	0,67	0,8
144P30FPS_HTTP_FLV_VLS_Download	0,93	0,97	0,95	*720P60FPS_WebRTC_VLS_Download*	0,96	0,88	0,92
144P30FPS_RTMP_VLS_Download	1	0,93	0,96	*1080P30FPS_DASH_VLS_Download*	0,96	1	0,98
144P30FPS_RTMP_VLS_Upload	0,93	0,93	0,93	*1080P30FPS_HLS_VLS_Download*	0,96	0,99	0,98
144P30FPS_WebRTC_VLS_Download	0,9	0,99	0,94	*1080P30FPS_HTTP_FLV_VLS_Download*	0,87	0,93	0,9
240P30FPS_DASH_VLS_Download	0,88	0,95	0,91	*1080P30FPS_RTMP_VLS_Download*	0,96	0,94	0,95
240P30FPS_HLS_VLS_Download	0,59	0,83	0,69	*1080P30FPS_RTMP_VLS_Upload*	0,92	0,92	0,92
240P30FPS_HTTP_FLV_VLS_Download	0,88	0,89	0,88	*1080P30FPS_WebRTC_VLS_Download*	0,91	0,78	0,84
240P30FPS_RTMP_VLS_Download	0,95	0,9	0,93	*1080P60FPS_DASH_VLS_Download*	0,88	0,96	0,92
240P30FPS_RTMP_VLS_Upload	0,97	0,9	0,94	*1080P60FPS_HLS_VLS_Download*	1	0,85	0,92
240P30FPS_WebRTC_VLS_Download	0,91	0,91	0,91	*1080P60FPS_HTTP_FLV_VLS_Download*	0,97	0,9	0,94
360P30FPS_DASH_VLS_Download	0,96	0,62	0,75	*1080P60FPS_RTMP_VLS_Download*	0,94	0,96	0,95
360P30FPS_HLS_VLS_Download	0,44	0,47	0,46	*1080P60FPS_RTMP_VLS_Upload*	0,76	0,96	0,85
360P30FPS_HTTP_FLV_VLS_Download	0,85	0,91	0,88	*1080P60FPS_WebRTC_VLS_Download*	0,87	0,87	0,87
360P30FPS_RTMP_VLS_Download	0,96	0,96	0,96	*1440P30FPS_DASH_VLS_Download*	0,95	0,88	0,91
360P30FPS_RTMP_VLS_Upload	0,97	0,93	0,95	*1440P30FPS_HLS_VLS_Download*	0,78	0,82	0,8
360P30FPS_WebRTC_VLS_Download	0,82	0,94	0,88	*1440P30FPS_HTTP_FLV_VLS_Download*	0,82	0,97	0,89
480P30FPS_DASH_VLS_Download	0,82	0,82	0,82	*1440P30FPS_RTMP_VLS_Download*	0,85	0,93	0,89
480P30FPS_HLS_VLS_Download	0,79	0,6	0,68	*1440P30FPS_RTMP_VLS_Upload*	0,97	0,79	0,87
480P30FPS_HTTP_FLV_VLS_Download	0,96	0,73	0,83	*1440P30FPS_WebRTC_VLS_Download*	0,95	0,88	0,91
480P30FPS_RTMP_VLS_Download	0,97	0,92	0,95	*1440P60FPS_DASH_VLS_Download*	0,95	0,81	0,88
480P30FPS_RTMP_VLS_Upload	0,99	0,9	0,94	*1440P60FPS_HLS_VLS_Download*	0,98	1	0,99
480P30FPS_WebRTC_VLS_Download	0,89	0,96	0,92	*1440P60FPS_HTTP_FLV_VLS_Download*	0,96	0,93	0,95
540P30FPS_DASH_VLS_Download	0,96	0,95	0,95	*1440P60FPS_RTMP_VLS_Download*	0,96	0,9	0,93
540P30FPS_HLS_VLS_Download	1	1	1	*1440P60FPS_RTMP_VLS_Upload*	0,94	0,94	0,94
540P30FPS_HTTP_FLV_VLS_Download	0,94	0,99	0,96	*1440P60FPS_WebRTC_VLS_Download*	0,94	0,91	0,93
540P30FPS_RTMP_VLS_Download	0,98	1	0,99	*2160P30FPS_DASH_VLS_Download*	0,89	1	0,94
540P30FPS_RTMP_VLS_Upload	0,89	0,92	0,91	*2160P30FPS_HLS_VLS_Download*	0,67	0,66	0,67
540P30FPS_WebRTC_VLS_Download	0,91	0,98	0,94	*2160P30FPS_HTTP_FLV_VLS_Download*	0,99	0,84	0,91
720P30FPS_DASH_VLS_Download	0,85	0,98	0,91	*2160P30FPS_RTMP_VLS_Download*	0,88	0,92	0,9
720P30FPS_HLS_VLS_Download	0,95	0,96	0,96	*2160P30FPS_RTMP_VLS_Upload*	0,86	0,96	0,91
720P30FPS_HTTP_FLV_VLS_Download	0,78	0,99	0,87	*2160P30FPS_WebRTC_VLS_Download*	0,91	0,88	0,89
720P30FPS_RTMP_VLS_Download	0,92	0,92	0,92	*2160P60FPS_DASH_VLS_Download*	0,9	0,93	0,92
720P30FPS_RTMP_VLS_Upload	0,78	0,96	0,86	*2160P60FPS_HLS_VLS_Download*	0,68	0,78	0,73
720P30FPS_WebRTC_VLS_Download	0,84	0,89	0,87	*2160P60FPS_HTTP_FLV_VLS_Download*	0,95	0,99	0,97
720P60FPS_DASH_VLS_Download	0,97	0,78	0,87	*2160P60FPS_RTMP_VLS_Download*	0,98	0,89	0,93
720P60FPS_HLS_VLS_Download	0,99	1	0,99	*2160P60FPS_RTMP_VLS_Upload*	0,93	0,94	0,93
720P60FPS_HTTP-FLV_VLS_Download	0,96	0,84	0,9	*2160P60FPS_WebRTC_VLS_Download*	0,94	1	0,97

3.2. Plūsmas līmeņa klasifikācijas veiktspēja

Plūsmas līmeņa klasifikācijas rezultātu apkopojums tiešraides straumēšanas datu kopai dots 10. tabulā. Kopējā klasifikācijas veiktspēja neapstrādātajiem datiem bija zema – augstākā sasniegtā precizitāte bija ierobežota ar lielumu 0,4 modelim *1D-CNN*. Tas lielā mērā skaidrojams ar neproporcionālu datu sadalījumu pēc kategorijām. Izpētot atsevišķas kategorijas, tika konstatēts, ka *360P_30FPS_VoD_YouTube* datu plūsma neapstrādātajā *PCAP* failā galvenokārt tiek pārsūtīta, izmantojot *QUIC* protokolu pāri *UDP*. Tajā ir četras informācijas apmaiņas plūsmas, kas atbilst četriem dinamiskajiem transporta slāņa portiem. Katra plūsma ilgst vairāk nekā 200 s, un viena no tām pat pārsniedz 800 s ilgumā. Līdzīgi, *480P_30FPS_VoD_YouTube* datu plūsma arī izmanto *UDP* protokolu, taču satur tikai divas informācijas apmaiņas plūsmas, kas nozīmē, ka sesijas laikā multivides sūtītājs portus mainījis tikai divreiz (30 minūšu sesijas ilgumā). Izmantojot rīku *NFStream*, jaunais plūsmas paraugs tiek izveidots, vai nu mainoties kādam no pieciem atribūtiem, vai arī pēc vienas minūtes taimera apstāšanās. Ja plūsma tiek pārtverta ilgāk par vienu minūti, tā tiek sadalīta jaunajos paraugos.

Šī iemesla dēļ gan *360P_30FPS_VoD_YouTube*, gan *480P_30FPS_VoD_YouTube* kategorijā tiek ģenerēti tikai 81 laikā ierobežoti plūsmas paraugi. Ievērojot to, ka tās pieci atribūti reti mainās dinamiskajā veidā, tiek efektīvi ierobežota plūsmas līmeņa paraugu ģenerēšanas efektivitāte. Pretstatā tam *720P_60FPS_VLS_Twitch* kategorija izmanto *TCP* protokolu un satur vairāk par 12 *TCP* savienojumiem, kas pārraida vairāk nekā 2000 plūsmas līmeņa paraugus. Visas trīs kategorijās plūsmas ilgst aptuveni 30 min, taču citi faktori, piemēram, *IP* adrešu maiņa, portu pārslēgšana vai *CDN* noslodzes balansēšana, var būtiski ietekmēt plūsmas paraugu sadalījumu.

Jāatzīmē, ka *Tor_Vimeo* neapstrādātā *PCAP* faila laika ilgums pārsniedz 3500 s, kas ir vairāk nekā divās reizēs ilgāk par pārējām datu kopām. Lai gan tajā ir tikai viena savienojuma plūsma, pavisam tiek ģenerēti vairāk nekā 3000 plūsmas līmeņa paraugi. Galvenokārt tas ir pamatojams ar to, ka *Tor* apvieno visas plūsmas vienā šifrētā plūsmā, kas tiek raidīta izveidotajā tunelī, kas tiek novirzīta uz sākotnējo aizsargāto mezglu (*guard relay*), nevis izveido atsevišķus savienojumus katrai plūsmai. Tas arī liecina par augstas kvalitātes video atskaņošanu ar īsiem intervāliem starp paketēm (*inter-arrival times – IAT*) un nozīmē, ka vienas sekundes plūsmas konteinerī nevar ievietot daudz pakešu vai datu, tādēļ nepieciešams lielāks konteineru skaits, kas savukārt dod lielāku plūsmas paraugu skaitu. Noslēgumā – plūsmas paraugu apjomu būtiski ietekmē pārsūtīšanas protokols un neapstrādātā *PCAP* faila izmērs. Pēc *SMOTEENN* lietošanas kategorijas *360P_30FPS_VoD_YouTube* un *480P_30FPS_VoD_YouTube* tika papildinātās ar jaunajiem paraugiem vairāk nekā desmitkārtīgajā apjomā, katrā veidojot kopumā vairāk par 1000 paraugiem, savukārt sākotnējās lielā apjoma kategorijas palika nemainīgas. Tas būtiski samazināja paraugu nevienmērīgu sadalījumu pēc klasēm (kategorijām) un uzlaboja klasifikācijas veiktspēju. Īpaši jāizceļ, ka slāņu modeļa makro *F1* rādītājs pārsniedza 0,99, pārspējot *1D-CNN* par 2 %. Saskaņā ar 11. tabulas datiem lielākā daļa kategoriju paraugu tika klasificēta pareizi, un daudzajām kategorijām tika sasniegta 100 % precizitāte. Viszemākais *F1* rādītājs bija 0,94, kas apliecina slāņu modeļa noturību un augsto veiktspēju.

Metavides datu kopai kopējā klasifikācijas veiktspēja ir salīdzinoši zema gan neapstrādātajos datos, gan pēc *SMOTEENN* lietošanas. Saskaņā ar 12. tabulas rezultātiem maksimālā klasifikācijas precizitāte pēc paraugu papildināšanas sasniedz 0,85, bet makro *F1* rādītājs – 0,73. Katras kategorijas neapstrādātā *PCAP* faila datu plūsmas ilgums ir ierobežots ar vienu minūti. Tā kā metavides lietotnes darbojās ar protokolu *WebRTC* un tiek pārsūtītas ar *TCP* protokola starpniecību, katrā *PCAP* failā transporta līmenī ir trīs savienojuma plūsmas. Tas rada aptuveni 180 paraugus plūsmas līmenī katrai kategorijai. Ņemot vērā nelielo kopējo paraugu skaitu, *SMOTE* nevar efektīvi ģenerēt adekvātus sintētiskos paraugus, un *ENN* ir tendence agresīvi dzēst liekos datus. Tas ir sevišķi problēmatiski tādās datu kopās, kurām ir līdzsvarots paraugu sadalījums pēc kategorijām, bet ir ievērojami trokšņi vai kategoriju pārklāšanās. Rezultātā pēc *SMOTEENN* katrā kategorijā paraugu skaits paliek mazāks par 10, un tas nav pietiekams apjoms, lai veiktu klasifikāciju efektīvi. 13. tabulā redzams, ka vairāk nekā piecas kategorijas iztrūkst no klasifikācijas rezultātiem. Tas ir saistīts ar pārbaudes kopas nelielo apjomu un izraisa to, ka dažas kategorijas netiek pārstāvētas pietiekamā apjomā testēšanas laikā, kā arī ir augsts kļūdaini klasificēto gadījumu skaits. Lielākoties, kategoriju *F1* rādītāji nepārsniedz 0,8. Šādi secinājumi apstiprina, ka neliela apjoma datu kopas nav

piemērotas detalizētajai klasifikācijai ne laika rindu līmenī, ne plūsmas līmenī. Lai nodrošinātu jēgpilnu klasifikācijas veiktspēju, ir nepieciešams atbilstošs datu apjoms.

Saskaņā ar tiešraides video straumēšanas simulācijas datu kopas klasifikācijas veiktspējas rezultātiem, kas ir apkopoti 14. tabulā, kopējie klasifikācijas rezultāti uzrāda būtisku uzlabojumu. Slāņu modelis sasniedza augstāko klasifikācijas precizitāti, 78 kategorijās pārspējot 0,9, pat bez paraugu papildināšanas. Šādu ievērojamu sniegumu nodrošināja līdzvērtīgais aptuveni 600 s ilgums visiem neapstrādātajiem *PCAP* datu plūsmu failiem un relatīvi līdzsvarotā paraugu sadalīšana kategorijās. Tomēr atšķirības straumēšanas protokolos joprojām ietekmē ģenerēto plūsmu paraugu skaitu. Piemēram, izmantojot *1080P30FPS* video ar *DASH* protokolu, kas balstīts *TCP* un *HTTP*, veidojās vairāk nekā 25 *TCP* savienojuma plūsmas. *HLS* protokols, kas arī ir balstīts *TCP* un *HTTP*, veido vairāk nekā 642 *TCP* savienojuma plūsmās. Savukārt *HTTP-FLV* un *RTMP* protokoli, kas izmanto pastāvīgas sesijas *TCP* savienojumus, katrs ģenerē tikai vienu *TCP* plūsmu. *WebRTC*, kas ir reāllaika protokols uz *UDP* protokola bāzes, veido tikai vienu *UDP* savienojuma plūsmu. *DASH* un *HLS* ir segmentēti un klienta pieprasījumos balstīti protokoli, kuros klients atkārtoti lejupielādē nelielus multivides segmentus un atskaņošanas saraksta failus, tādējādi veidojot daudz *TCP* savienojumu. *DASH* katram divu līdz 10 sekunžu video segmentam veic jaunu *HTTP* pieprasījumu, savukārt *HLS* bieži pieprasa atskaņošanas saraksta un segmentu failus. Salīdzinājumā, *HTTP-FLV* un *RTMP* uztur ilgstošas *TCP* sesijas bez biežām jauna savienojuma veikšanās nepieciešamībām. *WebRTC* izmanto pastāvīgu *UDP* plūsmu nepārtrauktajai pārraidei. Visām kategorijām, kuru datu plūsmas ilgums ir aptuveni 600 s, lielākā daļa ģenerē aptuveni 600 plūsmu paraugus. Tomēr *DASH* un *HLS* ģenerē vairāk plūsmu nekā citiem protokoli, pārsniedzot 10 % pieaugumu. Īpaši jāizceļ divas kategorijas – *2160P30FPS_HLS_VLS_Download* un *2160P60FPS_HLS_VLS_Download* – ar vairāk nekā 1500 *TCP* savienojumu plūsmām, un tā rezultātā ir vairāk nekā 2000 plūsmas līmeņa paraugi. Gadījumā ar *2160P60FPS_HLS_VLS_Download* video dekodēšanas ieviestā aizture (latentums) pagarināja atskaņošanas ilgumu līdz vairāk par 1500 s. Tas laika intervālus starp paketēm, un tā rezultātā plūsmu segmentēšanas process ģenerēja vairāk plūsmu paraugus, lai gan arī faktiskās pakešu skaits bija praktiski nemainīgs. Šāda laika nobīde ievieš trokšņus datos, kas negatīvi ietekmē klasifikācijas precizitāti laika rindas līmenī.

Pēc *SMOTEENN* lietošanas visu kategoriju paraugu skaits aptuveni divkāršojās. Šādai paplašinātai datu kopai slāņu modelis sasniedza klasifikācijas precizitāti 0,98 un makro *F1* rādītāju 0,97. Palielinātais plūsmu paraugu apjoms un daudzveidība ļāva arī citiem modeļiem sasniegt labāku veiktspēju. Piemēram, *1D-CNN* modelis ieguva makro *F1* rādītāju 0,87, kas ir apmēram par 10 % zemāks nekā slāņu modelim. Citi modeļi, piemēram, *RNN*, *Deep Packet*, *FlowPic* un *FS-Net* arī uzrādīja labus rezultātus, tomēr katrs atpalika no slāņu modeļa par vairāk nekā 20 %. *RNN* modelis uzrādīja visvājākos rezultātus ar klasifikācijas precizitāti apmēram 0,64 un makro *F1* rādītāju 0,57. Kā redzams 15. tabulā, slāņu modelis visām kategorijām saglabāja *F1* rādītājus tuvu 0,9, demonstrējot izcilu noturību un efektivitāti plūsmas līmeņa klasifikācijā. Veicot secinājumus, kopējā klasifikācijas veiktspēja visās trīs datu kopās ir izcila pēc paraugu papildināšanas, izmantojot 86 plūsmas līmeņa mācību pazīmes. Tomēr nelielais

paraugu apjoms joprojām ir pastāvīga problēma ar *NFStream* eksportēto plūsmu datiem, un nepieciešams izvērtēt papildu rīkus plūsmu pārvaldībai.

Tabula 10.

Klasifikācijas veiktspējas kopsavilkums tiešraides straumēšanas datu kopām plūsmas līmenī

Pārmērošana	Modelis	Kategorija	*P. W*	*R. W*	*F. A*	*F. I*	*A. Test*	*A. Train*	*A. Validate*
/	*1D-CNN*	86	0,45	0,44	0,39	0,44	0,43	0,56	0,55
/	*RNN*	86	0,30	0,33	0,26	0,31	0,30	0,34	0,34
/	*GCN*	86	0,34	0,35	0,31	0,35	0,35	0,51	0,35
/	*Stacking*	86	0,34	0,34	0,30	0,34	0,34	0,91	0,34
/	*Deep Packet*	86	0,34	0,34	0,30	0,34	0,35	0,40	0,38
/	*FlowPic*	86	0,29	0,31	0,27	0,31	0,29	0,33	0,34
/	*FS-Net*	86	0,33	0,33	0,29	0,33	0,34	0,39	0,39
SMOTEENN	*1D-CNN*	86	0,98	0,98	0,97	0,98	0,98	0,99	0,99
SMOTEENN	*RNN*	86	0,95	0,94	0,93	0,94	0,94	0,96	0,96
SMOTEENN	*GCN*	86	0,89	0,89	0,89	0,89	0,89	0,87	0,87
SMOTEENN	*Stacking*	86	0,99	0,99	0,99	0,99	0,99	1,00	0,99
SMOTEENN	*Deep Packet*	86	0,93	0,93	0,92	0,93	0,93	0,94	0,93
SMOTEENN	*FlowPic*	86	0,87	0,87	0,84	0,87	0,87	0,87	0,87
SMOTEENN	*FS-Net*	86	0,96	0,96	0,95	0,96	0,96	0,96	0,97

Tabula 11.

Klasifikācijas veiktspēja katrā tiešraides straumēšanas datu kopas kategorijā, izmantojot slāņu modeli pēc paraugu papildināšanas plūsmas līmenī

Kategorija	*P.*	*R.*	*F.*	Kategorija	*P.*	*R.*	*F.*
480P_3Mbps_N3_10Mhz	1	1	1	*360P_30FPS_VoD_Twitch*	0,99	1	1
480P_3Mbps_N78_10Mhz	0,99	0,99	0,99	*360P_30FPS_VLS_Twitch*	1	1	1
720P_6Mbps_N3_10Mhz	1	0,99	1	*480P_30FPS_VoD_Twitch*	1	1	1
1080P_10Mbps_N78_10Mhz	0,99	1	1	*480P_30FPS_VLS_Twitch*	1	1	1
2160P_30FPS_VoD_BiliBili	1	1	1	*720P_30FPS_VoD_Twitch*	1	1	1
2160P_60FPS_VoD_BiliBili	0,99	1	1	*720P_60FPS_VoD_Twitch*	1	1	1
2160P_120FPS_VoD_BiliBili	1	1	1	*720P_30FPS_VLS_Twitch*	1	1	1
2160P_60FPS_VLS_BiliBili	1	1	1	*720P_60FPS_VLS_Twitch*	1	1	1
2160P_80FPS_VLS_BiliBili	1	1	1	*1080P_30FPS_VoD_Twitch*	1	1	1
4320P_30FPS_VoD_BiliBili	1	0,99	1	*1080P_60FPS_VoD_Twitch*	1	1	1
360P_30FPS_VoD_BiliBili	0,99	0,99	0,99	*1080P_30FPS_VLS_Twitch*	1	1	1
480P_30FPS_VoD_BiliBili	0,99	1	0,99	*1080P_60FPS_VLS_Twitch*	1	1	1
480P_30FPS_VLS_BiliBili	1	1	1	*240P_30FPS_VoD_Vimeo*	0,98	0,99	0,98
720P_30FPS_VoD_BiliBili	0,99	0,99	0,99	*360P_30FPS_VoD_Vimeo*	0,96	0,97	0,96
720P_30FPS_VLS_BiliBili	1	0,99	1	*540P_30FPS_VoD_Vimeo*	0,97	0,97	0,97
1080P_30FPS_VoD_BiliBili	0,99	0,98	0,99	*720P_30FPS_VoD_Vimeo*	0,99	0,96	0,97
1080P_60FPS_VoD_BiliBili	1	0,99	1	*1080P_30FPS_VoD_Vimeo*	1	1	1
1080P_60FPS_VLS_BiliBili	0,99	1	1	*1440P_30FPS_VoD_YouTube*	0,99	0,99	0,99
1440P_60FPS_VLS_BiliBili	1	1	1	*1440P_60FPS_VoD_YouTube*	0,92	0,96	0,94
144P_30FPS_VOD_Facebook	1	0,99	0,99	*1440P_30FPS_VLS_YouTube*	1	0,99	0,99
144P_30FPS_VLS_Facebook	0,99	0,98	0,98	*1440P_60FPS_VLS_YouTube*	1	1	1
240P_30FPS_VLS_Facebook	0,96	0,99	0,97	*2160P_30FPS_VoD_YouTube*	1	0,99	1
360P_30FPS_VOD_Facebook	0,98	0,99	0,99	*2160P_60FPS_VoD_YouTube*	0,97	0,91	0,94
360P_30FPS_VLS_Facebook	0,95	0,95	0,95	*2160P_30FPS_VLS_YouTube*	0,99	1	0,99
480P_30FPS_VOD_Facebook	0,99	1	1	*2160P_60FPS_VLS_YouTube*	1	1	1
480P_30FPS_VLS_Facebook	0,97	0,96	0,97	*4320P_30FPS_VoD_YouTube*	1	1	1
540P_30FPS_VOD_Facebook	1	1	1	*4320P_60FPS_VoD_YouTube*	1	1	1
720P_30FPS_VOD_Facebook	1	1	1	*144P_30FPS_VoD_YouTube*	0,99	1	1
720P_30FPS_VLS_Facebook	1	1	1	*144P_30FPS_VLS_YouTube*	0,99	1	1
1080P_30FPS_VOD_Facebook	1	1	1	*240P_30FPS_VoD_YouTube*	0,99	0,99	0,99
1080P_30FPS_VLS_Facebook	1	1	1	*240P_30FPS_VLS_YouTube*	1	0,99	1
360P_30FPS_VLS_TikTok	1	1	1	*360P_30FPS_VoD_YouTube*	1	1	1
540P_30FPS_VLS_TikTok	1	1	1	*360P_30FPS_VLS_YouTube*	1	1	1

11. tabulas turpinājums

720P_30FPS_VLS_TikTok	1	1	1	*480P_30FPS_VoD_YouTube*	1	1	1
720P_60FPS_VLS_TikTok	1	1	1	*480P_30FPS_VLS_YouTube*	1	1	1
1080P_30FPS_VLS_TikTok	1	1	1	*720P_30FPS_VoD_YouTube*	1	1	1
1080P_60FPS_VLS_TikTok	1	1	1	*720P_60FPS_VoD_YouTube*	1	1	1
Tor_Vimeo	1	1	1	*720P_30FPS_VLS_YouTube*	0,94	0,96	0,95
Tor_YouTube	1	1	1	*720P_60FPS_VLS_YouTube*	0,99	0,99	0,99
VPN_Vimeo	1	0,99	0,99	*1080P_30FPS_VoD_YouTube*	1	1	1
VPN_YouTube	0,99	1	0,99	*1080P_60FPS_VoD_YouTube*	0,99	0,99	0,99
160P_30FPS_VoD_Twitch	1	1	1	*1080P_30FPS_VLS_YouTube*	0,97	0,95	0,96
160P_30FPS_VLS_Twitch	1	1	1	*1080P_60FPS_VLS_YouTube*	0,99	1	1

Tabula 12.

Klasifikācijas veiktspējas kopsavilkums metavides datu kopām plūsmas līmenī

Pārmērošana	**Modelis**	**Kategorija**	***P. W***	***R. W***	***F. A***	***F. I***	***A. Test***	***A. Train***	***A. Validate***
/	*1D-CNN*	21	0,49	0,45	0,45	0,45	0,45	0,56	0,57
/	*RNN*	21	0,31	0,39	0,34	0,34	0,29	0,39	0,33
/	*GCN*	21	0,38	0,35	0,35	0,35	0,35	0,43	0,36
/	*Stacking*	21	0,47	0,46	0,45	0,46	0,46	1,00	0,48
/	*Deep Packet*	21	0,37	0,36	0,37	0,36	0,37	0,40	0,40
/	*FlowPic*	21	0,32	0,30	0,31	0,30	0,30	0,35	0,32
/	*FS-Net*	21	0,40	0,37	0,36	0,37	0,36	0,43	0,40
SMOTEENN	*1D-CNN*	21	0,87	0,87	0,73	0,85	0,84	0,87	0,89
SMOTEENN	*RNN*	21	0,63	0,67	0,40	0,63	0,60	0,68	0,71
SMOTEENN	*GCN*	21	0,80	0,77	0,66	0,77	0,77	0,94	0,80
SMOTEENN	*Stacking*	21	0,54	0,53	0,51	0,53	0,53	1,00	0,51
SMOTEENN	*Deep Packet*	21	0,64	0,66	0,39	0,60	0,61	0,62	0,58
SMOTEENN	*FlowPic*	21	0,49	0,66	0,33	0,58	0,46	0,51	0,47
SMOTEENN	*FS-Net*	21	0,79	0,80	0,53	0,76	0,72	0,80	0,79

Tabula 13.

Klasifikācijas veiktspēja katrā metavides datu kopas kategorijā, izmantojot 1D-CNN modeli pēc paraugu papildināšanas plūsmas līmenī

Kategorija	***P.***	***R.***	***F.***
BigscreenTheatre_60FPS	0,75	0,6	0,67
BigscreenTheatre_90FPS	0,75	0,9	0,82
BigscreenTheatre_120FPS	0,5	1	0,67
DiRTRally2.0_60FPS	1	0,5	0,67
DiRTRally2.0_90FPS	0,6	1	0,75
DiRTRally2.0_120FPS	0	0	0
Hellblade_60FPS	1	0,5	0,67
Hellblade_90FPS	0	0	0
Hellblade_120FPS	0,67	1	0,8
RealityMixer+SteamVRHome_60FPS	0,67	1	0,8
RealityMixer+SteamVRHome_90FPS	0,89	0,73	0,8
RealityMixer+SteamVRHome_120FPS	0,75	0,6	0,67
SolarSystemAR_60FPS	0,2	0,33	0,25
SolarSystemAR_90FPS	0	0	0
SolarSystemAR_120FPS	1	1	1
TheLab_60FPS	0,73	0,73	0,73
TheLab_90FPS	0,67	1	0,8
TheLab_120FPS	0,67	0,8	0,73
VRChat_60FPS	1	0,75	0,86
VRChat_90FPS	0	0	0
VRChat_120FPS	0	0	0

Tabula 14.

Klasifikācijas veiktspējas kopsavilkums tiešraides video straumēšanas datu kopām plūsmas līmenī

Pārmērošana	Modelis	Kategorija	*P. W*	*R. W*	*F. A*	*F. I*	*A. Test*	*A. Train*	*A. Validate*
/	*1D-CNN*	78	0,74	0,73	0,73	0,73	0,73	0,78	0,77
/	*RNN*	78	0,68	0,67	0,66	0,67	0,67	0,69	0,69
/	*GCN*	78	0,59	0,58	0,55	0,58	0,58	0,58	0,58
/	*Stacking*	78	0,90	0,90	0,89	0,90	0,90	0,98	0,90
/	*Deep Packet*	78	0,68	0,68	0,67	0,68	0,67	0,68	0,68
/	*FlowPic*	78	0,65	0,64	0,63	0,64	0,64	0,65	0,65
/	*FS-Net*	78	0,72	0,71	0,71	0,71	0,71	0,72	0,71
SMOTEENN	*1D-CNN*	78	0,91	0,90	0,87	0,90	0,90	0,92	0,92
SMOTEENN	*RNN*	78	0,84	0,84	0,78	0,84	0,84	0,84	0,84
SMOTEENN	*GCN*	78	0,70	0,64	0,57	0,64	0,64	0,69	0,69
SMOTEENN	*Stacking*	78	0,98	0,98	0,97	0,98	0,98	1,00	0,98
SMOTEENN	*Deep Packet*	78	0,82	0,82	0,75	0,82	0,82	0,82	0,83
SMOTEENN	*FlowPic*	78	0,79	0,78	0,71	0,78	0,78	0,78	0,79
SMOTEENN	*FS-Net*	78	0,87	0,86	0,81	0,86	0,86	0,86	0,86

Tabula 15.

Klasifikācijas veiktspēja katrā tiešraides video straumēšanas datu kopas kategorijā, izmantojot slāņu modeli pēc paraugu papildināšanas plūsmas līmenī

Kategorija	P.	R.	F.	Kategorija	P.	R.	F.
144P30FPS_DASH_VLS_Download	0,98	0,97	0,98	*720P60FPS_RTMP_VLS_Download*	1	1	1
144P30FPS_HLS_VLS_Download	0,9	0,9	0,9	*720P60FPS_RTMP_VLS_Upload*	1	1	1
144P30FPS_HTTP_FLV_VLS_Download	1	1	1	*720P60FPS_WebRTC_VLS_Download*	0,99	1	1
144P30FPS_RTMP_VLS_Download	0,99	1	1	*1080P30FPS_DASH_VLS_Download*	0,97	0,95	0,96
144P30FPS_RTMP_VLS_Upload	1	1	1	*1080P30FPS_HLS_VLS_Download*	0,95	0,98	0,97
144P30FPS_WebRTC_VLS_Download	1	1	1	*1080P30FPS_HTTP_FLV_VLS_Download*	1	1	1
240P30FPS_DASH_VLS_Download	0,97	0,97	0,97	*1080P30FPS_RTMP_VLS_Download*	0,99	1	1
240P30FPS_HLS_VLS_Download	0,89	0,85	0,87	*1080P30FPS_RTMP_VLS_Upload*	1	1	1
240P30FPS_HTTP_FLV_VLS_Download	1	1	1	*1080P30FPS_WebRTC_VLS_Download*	0,91	0,92	0,91
240P30FPS_RTMP_VLS_Download	1	0,99	1	*1080P60FPS_DASH_VLS_Download*	0,92	0,95	0,93
240P30FPS_RTMP_VLS_Upload	1	1	1	*1080P60FPS_HLS_VLS_Download*	0,88	0.84	0,86
240P30FPS_WebRTC_VLS_Download	1	0,99	0,99	*1080P60FPS_HTTP_FLV_VLS_Download*	1	1	1
360P30FPS_DASH_VLS_Download	1	0,98	0,99	*1080P60FPS_RTMP_VLS_Download*	1	0.99	0,99
360P30FPS_HLS_VLS_Download	0,85	0,88	0,87	*1080P60FPS_RTMP_VLS_Upload*	1	1	1
360P30FPS_HTTP_FLV_VLS_Download	1	1	1	*1080P60FPS_WebRTC_VLS_Download*	0,92	0,85	0,88
360P30FPS_RTMP_VLS_Download	1	1	1	*1440P30FPS_DASH_VLS_Download*	0,95	0,93	0,94
360P30FPS_RTMP_VLS_Upload	1	1	1	*1440P30FPS_HLS_VLS_Download*	0,81	0,86	0,84
360P30FPS_WebRTC_VLS_Download	0,98	0,99	0,98	*1440P30FPS_HTTP_FLV_VLS_Download*	1	1	1
480P30FPS_DASH_VLS_Download	0,98	0,97	0,98	*1440P30FPS_RTMP_VLS_Download*	1	1	1
480P30FPS_HLS_VLS_Download	0,89	0,89	0,89	*1440P30FPS_RTMP_VLS_Upload*	1	1	1
480P30FPS_HTTP_FLV_VLS_Download	1	1	1	*1440P30FPS_WebRTC_VLS_Download*	0,93	0,93	0,93
480P30FPS_RTMP_VLS_Download	1	0,99	0,99	*1440P60FPS_DASH_VLS_Download*	0,94	0,95	0,94
480P30FPS_RTMP_VLS_Upload	1	1	1	*1440P60FPS_HLS_VLS_Download*	0,82	0,88	0,85
480P30FPS_WebRTC_VLS_Download	1	1	1	*1440P60FPS_HTTP_FLV_VLS_Download*	1	1	1
540P30FPS_DASH_VLS_Download	0,96	0,98	0,97	*1440P60FPS_RTMP_VLS_Download*	1	1	1
540P30FPS_HLS_VLS_Download	0,98	0,96	0,97	*1440P60FPS_RTMP_VLS_Upload*	1	1	1
540P30FPS_HTTP_FLV_VLS_Download	1	1	1	*1440P60FPS_WebRTC_VLS_Download*	0,9	0,93	0,91
540P30FPS_RTMP_VLS_Download	1	1	1	*2160P30FPS_DASH_VLS_Download*	0,96	0,98	0,97
540P30FPS_RTMP_VLS_Upload	1	1	1	*2160P30FPS_HLS_VLS_Download*	0,88	0,82	0,85
540P30FPS_WebRTC_VLS_Download	0,92	0,97	0,94	*2160P30FPS_HTTP_FLV_VLS_Download*	1	1	1
720P30FPS_DASH_VLS_Download	0,98	0,96	0,97	*2160P30FPS_RTMP_VLS_Download*	1	0,99	1
720P30FPS_HLS_VLS_Download	1	1	1	*2160P30FPS_RTMP_VLS_Upload*	1	1	1
720P30FPS_HTTP_FLV_VLS_Download	1	1	1	*2160P30FPS_WebRTC_VLS_Download*	0,93	0,9	0,91
720P30FPS_RTMP_VLS_Download	0,99	1	1	*2160P60FPS_DASH_VLS_Download*	0,98	0,9	0,99
720P30FPS_RTMP_VLS_Upload	1	1	1	*2160P60FPS_HLS_VLS_Download*	0,9	0,87	0,89
720P30FPS_WebRTC_VLS_Download	0,99	0,98	0,98	*2160P60FPS_HTTP_FLV_VLS_Download*	1	1	1
720P60FPS_DASH_VLS_Download	0,96	0,97	0,97	*2160P60FPS_RTMP_VLS_Download*	1	1	1
720P60FPS_HLS_VLS_Download	1	1	1	*2160P60FPS_RTMP_VLS_Upload*	1	1	1
720P60FPS_HTTP-FLV_VLS_Download	1	1	1	*2160P60FPS_WebRTC_VLS_Download*	1	1	1

3.3. Vērtuma līmeņa klasifikācijas veiktspēja

Vērtuma (lietderīgo datu) līmeņa klasifikācijas uzdevumā visi baitu lietderīgās daļas paraugi tika pārveidoti grafu reprezentācijās. Rezultātā *SMOTEENN* paraugu papildināšanas metodes piemērošana nav iespējama, jo *SMOTE* un *ENN* metodes ir izstrādātas datiem, kuru formāts ir tabula. Šādas metodes pieņem, ka katrs paraugs ir fiksēta garuma vektors un attālumam starp paraugiem ir nozīme, turklāt parasti tiek lietota Eiklīda distance. Savukārt grafu datos, kur katram paraugam ir unikāla struktūra un to veido mezgli un malas, interpolācija starp paraugiem nav tieši izteikta. Tādēļ tādas metodes kā *SMOTE* un *ENN* nav lietojamas grafu datu kopām. Visos interaktīvās multimediju tīkla datu plūsmas vērtuma līmenī, pat kategorijā *144P_30FPS_DASH_VLS_Download* ar vismazāko *PCAP* pārtverto pakešu skaitu, ir pietiekami liels pakešu skaits, kas pārsniedz 5000 paketes. Tas nodrošina pietiekamu vērtuma paraugu apjomu, tādēļ sintētiskā datu papildināšana principiāli arī nav nepieciešama. Visas astoņas klasifikācijas metodes tika izvērtētas vērtuma līmenī. Neskaitot *2D-CNN* modeli, kas tieši darbojas ar oriģinālajām grafu struktūrām, visiem pārējie modeļiem bija jāpārveido grafus vienmērīgajos viendimensijas funkciju vektoros un jāpiemēro principiālo komponenšu analīzi (*PCA*) dimensiju skaita samazināšanai. Kā redzams 16. tabulā, *2D-CNN* modelis sasniedza augstāko klasifikācijas precizitāti – 0,81 pārbaudes datu kopai. Tomēr gan mācību, gan validācijas precizitāte bija 0,98, kas liecina par pārmērīgo pielāgošanu (*overfitting*). Modelis iemācījās saglabāt atmiņā mācību un validācijas datus, nevis meklēt vispārējās likumsakarības. Šī problēma ir izteiktāka *1D-CNN* modelī, kur starpība starp mācību un pārbaudes datu kopu klasifikācijas precizitāti pārsniedz 28 %.

Pārējie modeļi demonstrēja līdzīgus makro *F1* rādītājus, kas parasti bija ap 0,7. *2D-CNN* modelim makro *F1* rādītājs bija tuvs 0,82, kas ir par aptuveni 15 % augstāks rezultāts nekā citiem modeļiem. Lai gan datu kopā nav novērojama paraugu sadalījuma nevienmērīgums pēc kategorijām, būtiska problēma ir piemērotu paņēmienu izvēle pazīmju sintēzei (*feature fusion*). Nepietiekami efektīvas sintēzes stratēģijas neļauj modelim apgūt noderīgas klasifikācijas pazīmes vai definēt skaidras lēmumu robežas. 17. tabulā uzrādīts veiktspējas sadalījums pēc kategorijām. Piemēram, kategorijai *480P_30FPS_VLS_Facebook F1* rādītājs bija tikai 0,26, bet *720P_30FPS_VOD_Facebook* kategorijā tas bija vēl zemāks – 0,14. Abos gadījumos izmantotās pazīmju sintēzes metodes iekļāva gadījumrakstura pikseļu sajaukšanu un to nejaušu pārbīdi, kas neuzlaboja veiktspēju. Līdzīgi, kategorijām *360P_30FPS_VLS_TikTok* un *540P_30FPS_VLS_TikTok F1* rādītāji bija zem 0,4. Šie rezultāti tika iegūti, izmantojot tādus sintēzes paņēmienus kā gadījumrakstura Gausa aizmiglojums, mainīgā neliela kodola asuma palielināšana (*sharpening*), divdimensiju attēlu filtrēšana un nejaušu raksturlielumu izklāšanas pievienošana. Vairāk nekā 20 kategorijām *F1* rādītāji bija zemāki par 0,80. Tas liecina par to, ka stohastiskā trokšņa pievienošana vai nelielu grafu izmaiņu veikšana ar permutācijām vai kombinācijām nodrošina tikai ierobežotas klasifikācijas veiktspējas uzlabošanās.

Salīdzinot 18. un 12. tabulas datus, var apgalvot, ka metavides datu kopai klasifikācijas veiktspēja datu vērtuma līmenī ir būtiski precīzāka nekā plūsmas līmenī. *2D-CNN* modelis sasniedza klasifikācijas precizitāti 0,81, un makro *F1* rādītājs ir 0,82. Tomēr ir novērojamas pārmērīgās pielāgošanās pazīmes, jo apmācības un validācijas datu precizitāte ir ievērojami

augstāka, salīdzinot ar pārbaudes datu precizitāti. Neskatoties uz to, *2D-CNN* modelis kopumā pārspēja visus pārējos modeļus par vairāk nekā 10 % klasifikācijas precizitātes ziņā. Saskaņā ar 19. tabulu, kurā redzama klasifikācijas veiktspēja pēc atsevišķām kategorijām, tikai četrām kategorijām – *RealityMixer+SteamVRHome_90FPS*, *SolarSystemAR_60FPS*, *TheLab_60FPS* un *VRChat_60FPS* – *F1* rādītāji bija zemāki par 0,6. Pārējās kategorijas sasniedza *F1* rādītājus virs 0,8, liecinot par salīdzinoši augstu veiktspēju datu kopas lielākajai daļai. Klasifikācijas precizitātes pazemināšanā zemākās veiktspējas kategorijām būtiska nozīme ir vairākiem faktoriem. Tīkla oriģinālo datu plūsmas pārtveršanas laikā datu pārraides ātrums bija ierobežots ar 15 Mbps, un tika izmantots *WebRTC* straumēšanas protokols ar *TCP* starpniecību. Ievērojot to, ka tika atļauti *TSO* vai *GSO*, daudzi *TCP* segmenti tika saglabāti *PCAP* failos nefragmentētajā veidā un garajās secībās, turklāt atsevišķajos gadījumos lietderīgas daļas apjoms pārsniedza 10 000 baitus. Tomēr grafu paraugu ģenerēšanai tika izmantoti tikai katras paketes pirmie 1250 baiti, visi pārējie paketes lietderīgie dati tika izlaisti. Turklāt būtiskajai pakešu daļai lietderīgo datu apjoms nepārsniedza 100 baitus. Lai gan datu kopa ietvēra pietiekamu paraugu skaitu, atsevišķajiem grafu paraugiem bieži vien pietrūka informatīvas vai efektīvas pazīmes, kas apgrūtināja precīzu klasifikāciju. Jāatzīmē, ka daudzās kategorijās bija atkarīgas no tādu pazīmju sintēzes metožu lietošanās kā, piemēram, nejauša rotācija, vertikāla apgriešana, Gausa trokšņu pievienošana un Gausa aizmiglojums ar nejaušu kodola izmēru. Šie standarta datu papildināšanas (*augmentation*) paņēmieni nespēja izteikti uzlabot klasifikācijai būtiskās pazīmes, galvenokārt tāpēc, ka pašas oriģinālās pazīmes bija ierobežotas gan to kvalitātes, gan informatīvās vērtības ziņā.

Tiešraides video straumēšanas simulācijas datu kopai klasifikācijas veiktspēja ir samērā zema vērtuma līmenī. Kā redzams 20. tabulā, visaugstākā kopējā klasifikācijas precizitāte sasniedz tikai 0,75 modelim *2D-CNN* ar 78 kategorijām. Šis rādītājs ir vairāk nekā 20 % zemāks, salīdzinot ar klasifikācijas precizitāti laika rindas līmenī, kur *1D-CNN* modeļa precizitāte sasniedza 0,91, bet slāņu modeļa precizitāte plūsmas līmenī – 0,98. Jēldatu *PCAP* failu analīze atklāj atšķirības protokolos, kas ietekmē klasifikācijas veiktspēju. *WebRTC* protokolam, kura pamatā ir *UDP*, 80 % no pārraidāmo pakešu apjoma lietderīgas daļas lielums nepārsniedz 500 baitus. Rezultātā pietrūkst augstās kvalitātes efektīvo paraugu grafu veidošanai. Savukārt tādi protokoli kā *DASH*, *HLS*, *HTTP-FLV* un *RTMP* darbojas ar *TCP* protokola starpniecību. Šajos protokolos bieži sastopami ļoti lieli lietderīgo datu segmenti, kuru izmērs var pārsniegt 10 000 baitu un pat sasniegt 65 549 baitus *DASH* protokola gadījumā. Tomēr, ņemot vērā glabāšanas ierobežojumus (1250 baiti) grafu pārveidošanas laikā, tiek saglabāti tikai aptuveni 2 % no lietderīgo datu baitiem, un noderīgās informācijas lielākā daļa tiek pazaudēta. Turklāt vairāk nekā 0,5 % no kopējā pakešu skaita ietver lietderīgos datus ar apjomu zem 100 baitiem. Kā redzams 21. tabulā, lielākajai kategoriju daļai, kas saistītas ar *DASH* protokolu, *F1* rādītāji ir zem 0,5. Šīs problēmas rodas *TCP* noslodzes samazināšanas (*offloading*) mehānismu darbības dēļ, kad operētājsistēma un tīkla saskarnes karte efektivitātes nolūkos apvieno lielus *TCP* segmentus, kas turpmāk tiek sadalīti mazākās paketēs to pārraidīšanai. Veicot datu plūsmas pārtveršanu raidītāja pusē, sevišķi no atgriezeniskās saites (*loopback*) saskarnēm, šie lieli segmenti parādās *PCAP* failos pat tad, ja tie netika tieši pārraidīti tīklā. Savukārt protokoli, kuru pamatā ir *UDP*, piemēram, *WebRTC*, šos noslodzes samazināšanas mehānismus neatbalsta un

līdzīgu uzvedību neuzrāda. Metavides datu kopai *WebRTC* atsevišķajās situācijās ir nokonfigurēts tā, lai tiktu pārslēgts uz *TCP* protokolu, ņemot vērā tīkla ierobežojumus. Šāda atkāpe veido līdzīgus *TCP* segmentus ar lielu garumu gadījumos, kad ir atspējots kāds no *TSO* vai *GSO* mehānismiem. Lai gan *WebRTC* pamatā ir *UDP* protokols, reālas lietošanas scenārijos bieži tiek izmantoti arī *TCP* protokola atkāpes maršruti, kas izraisa tādas pašas noslodzes samazināšanas mehānismu sekas, kādas novērojamas *DASH* un *HLS* datu plūsmu gadījumā. Izmantojot 2000 nejauši izvēlētus paraugus, lietderīgās informācijas apjoma ierobežojumi mazā izmēra paketēs turpina ierobežotā lietderīgā informācija mazajās paketēs turpina traucēt pareizu klasifikāciju. Pat tad, ja tika lietotas grafu bāzētās pazīmju sintēzes metodes, rezultāti joprojām ir neapmierinoši, jo oriģinālo grafu paraugu kvalitāte nav pietiekama. Salīdzinot ar tiešraides straumēšanas datu kopu, *VLS* simulācijas datu kopa ietver lielāku *TCP* datu plūsmu apjomu, kā arī paraugi ar zemās kvalitātes lietderīgo daļu veido lielāku īpatsvaru no kopējās datu plūsmas.

Kopumā, klasifikācijas veiktspēja vērtuma līmenī ievērojami atšķiras dažādām interaktīvo mediju lietotnēm. Datu plūsmu tipu struktūru raksturojumu nesakritība apgrūtina modeļu vispārināšanu. Lai gan tika lietotas dažādas pazīmju sintēzes metodes ar mērķi iegūt kopīgas pazīmes, grafu paraugu ierobežotā informācija samazināja šo pasākumu efektivitāti. Datu kopas ģenerēšanas laikā tika izmantota papildināšana ar nullēm, lai papildinātu lietderīgos datus, kas ir nepietiekamajā daudzumā, taču tāda pieeja grafiem nepievienoja lietderīgo informāciju. Tādējādi standarta datu papildināšanas paņēmieni, piemēram, formas rotācija un nejaušo trokšņu pievienošana, nebija efektīvi klasifikācijas precizitātes uzlabošanā. Šīs metodes nav labi piemērotas datu struktūrai, un ir nepieciešams izpētīt progresīvākas pazīmju sintēzes stratēģijas.

Tabula 16.
Interaktīvo tiešraides datu kopu klasifikācijas veiktspējas kopsavilkums vērtuma līmenī

Modelis	**Kategorija**	***P. W***	***R. W***	***F. A***	***F. I***	***A. Test***	***A. Train***	***A. Validate***
2D-CNN	95	0,82	0,81	0,82	0,81	0,81	0,98	0,98
1D-CNN	95	0,71	0,71	0,71	0,71	0,71	0,99	0,99
RNN	95	0,73	0,73	0,72	0,73	0,73	0,77	0,78
GCN	95	0,70	0,69	0,68	0,69	0,69	0,70	0,68
Stacking	95	0,74	0,74	0,74	0,74	0,74	0,99	0,74
Deep Packet	95	0,73	0,72	0,72	0,72	0,72	0,79	0,80
FlowPic	95	0,72	0,71	0,70	0,71	0,71	0,73	0,73
FS-Net	95	0,75	0,74	0,74	0,74	0,74	0,77	0,77

Tabula 17.
Klasifikācijas veiktspēja katrā tiešraides straumēšanas datu kopas kategorijā, izmantojot 2D-CNN modeli vērtuma līmenī

Kategorija	*P.*	*R.*	*F.*	Kategorija	*P.*	*R.*	*F.*
480P_3Mbps_N3_10Mhz	0,83	0,69	0,75	*VPN_Vimeo*	0,99	0,98	0,98
480P_3Mbps_N78_10Mhz	1	1	1	*VPN_YouTube*	0,52	0,52	0,52
480P_3Mbps_N78_20Mhz	1	1	1	*160P_30FPS_VoD_Twitch*	0,99	0,98	0,99
720P_6Mbps_N3_10Mhz	1	1	1	*160P_30FPS_VLS_Twitch*	1	1	1
720P_6Mbps_N78_10Mhz	1	1	1	*360P_30FPS_VoD_Twitch*	0,48	0,47	0,48
720P_6Mbps_N78_20Mhz	1	1	1	*360P_30FPS_VLS_Twitch*	0,99	1	0,99
1080P_10Mbps_N3_10Mhz	1	0,98	0,99	*480P_30FPS_VoD_Twitch*	0,98	0,96	0,97
1080P_10Mbps_N78_10Mhz	1	1	1	*480P_30FPS_VLS_Twitch*	0,83	0,94	0,88
1080P_10Mbps_N78_20Mhz	0,95	0,99	0,97	*720P_30FPS_VoD_Twitch*	1	1	1
2160P_20Mbps_N3_10Mhz	0,79	0,92	0,85	*720P_60FPS_VoD_Twitch*	1	0,97	0,98
2160P_20Mbps_N78_20Mhz	1	0,99	0,99	*720P_30FPS_VLS_Twitch*	0,67	0,84	0,74
4320P_40Mbps_N3_10Mhz	0,99	0,99	0,99	*720P_60FPS_VLS_Twitch*	0,72	0,66	0,69
4320P_40Mbps_N78_20Mhz	0,99	0,99	0,99	*1080P_30FPS_VoD_Twitch*	0,95	1	0,98
2160P_30FPS_VoD_BiliBili	1	0,99	0,99	*1080P_60FPS_VoD_Twitch*	1	0,99	1
2160P_60FPS_VoD_BiliBili	1	1	1	*1080P_30FPS_VLS_Twitch*	0,92	0,79	0,85
2160P_120FPS_VoD_BiliBili	0,85	0,85	0,85	*1080P_60FPS_VLS_Twitch*	0,99	0,96	0,97
2160P_60FPS_VLS_BiliBili	0,69	0,67	0,68	*240P_30FPS_VoD_Vimeo*	0,74	0,8	0,77
2160P_80FPS_VLS_BiliBili	0,95	0,96	0,96	*360P_30FPS_VoD_Vimeo*	1	1	1
4320P_30FPS_VoD_BiliBili	0,87	0,87	0,87	*540P_30FPS_VoD_Vimeo*	0,49	0,49	0,49
360P_30FPS_VoD_BiliBili	0,9	0,89	0,89	*720P_30FPS_VoD_Vimeo*	0,34	0,38	0,36
480P_30FPS_VoD_BiliBili	0,95	0,87	0,91	*1080P_30FPS_VoD_Vimeo*	0,65	0,66	0,66
480P_30FPS_VLS_BiliBili	1	1	1	*1440P_30FPS_VoD_YouTube*	0,5	0,48	0,49
720P_30FPS_VoD_BiliBili	0,55	0,47	0,51	*1440P_60FPS_VoD_YouTube*	0,41	0,31	0,35
720P_30FPS_VLS_BiliBili	1	0,97	0,98	*1440P_30FPS_VLS_YouTube*	0,99	1	0,99
1080P_30FPS_VoD_BiliBili	1	1	1	*1440P_60FPS_VLS_YouTube*	0,26	0,29	0,27
1080P_60FPS_VoD_BiliBili	0,99	1	1	*2160P_30FPS_VoD_YouTube*	0,27	0,43	0,33
1080P_60FPS_VLS_BiliBili	0,89	0,94	0,92	*2160P_60FPS_VoD_YouTube*	0,81	0,68	0,74
1440P_60FPS_VLS_BiliBili	1	0,99	0,99	*2160P_30FPS_VLS_YouTube*	0,97	0,99	0,98
144P_30FPS_VOD_Facebook	0,98	1	0,99	*2160P_60FPS_VLS_YouTube*	0,99	0,99	0,99
144P_30FPS_VLS_Facebook	0,8	0,72	0,76	*4320P_30FPS_VoD_YouTube*	0,99	1	0,99
240P_30FPS_VLS_Facebook	0,97	0,99	0,98	*4320P_60FPS_VoD_YouTube*	0,96	0,94	0,95
360P_30FPS_VOD_Facebook	0,98	0,91	0,94	*144P_30FPS_VoD_YouTube*	0,78	1	0,88
360P_30FPS_VLS_Facebook	1	1	1	*144P_30FPS_VLS_YouTube*	0,97	1	0,98
480P_30FPS_VOD_Facebook	1	1	1	*240P_30FPS_VoD_YouTube*	1	1	1
480P_30FPS_VLS_Facebook	0,27	0,26	0,26	*240P_30FPS_VLS_YouTube*	1	1	1
540P_30FPS_VOD_Facebook	0,62	0,65	0,64	*360P_30FPS_VoD_YouTube*	1	0,89	0,94
720P_30FPS_VOD_Facebook	0,17	0,12	0,14	*360P_30FPS_VLS_YouTube*	0,96	0,86	0,91
720P_30FPS_VLS_Facebook	0,9	0,76	0,82	*480P_30FPS_VoD_YouTube*	0,83	0,75	0,79
1080P_30FPS_VOD_Facebook	0,77	0,87	0,82	*480P_30FPS_VLS_YouTube*	0,51	0,46	0,48
1080P_30FPS_VLS_Facebook	0,99	0,99	0,99	*720P_30FPS_VoD_YouTube*	0,34	0,4	0,37
360P_30FPS_VLS_TikTok	0,3	0,39	0,34	*720P_60FPS_VoD_YouTube*	1	0,99	1
540P_30FPS_VLS_TikTok	0,4	0,37	0,38	*720P_30FPS_VLS_YouTube*	0,99	0,99	0,99
720P_30FPS_VLS_TikTok	0,85	1	0,92	*720P_60FPS_VLS_YouTube*	0,72	0,63	0,67
720P_60FPS_VLS_TikTok	0,56	0,64	0,6	*1080P_30FPS_VoD_YouTube*	0,51	0,47	0,49
1080P_30FPS_VLS_TikTok	0,31	0,2	0,24	*1080P_60FPS_VoD_YouTube*	1	1	1
1080P_60FPS_VLS_TikTok	0,62	0,58	0,6	*1080P_30FPS_VLS_YouTube*	1	1	1
Tor_Vimeo	0,62	0,6	0,61	*1080P_60FPS_VLS_YouTube*	0,23	0,3	0,26
Tor_YouTube	1	0,9	0,95				

Tabula 18.
Klasifikācijas veiktspējas kopsavilkums metavides datu kopām vērtuma līmenī

Modelis	Kategorija	*P. W*	*R. W*	*F. A*	*F. I*	*A. Test*	*A. Train*	*A. Validate*
2D-CNN	21	0,84	0,84	0,85	0,84	0,84	1,00	1,00
1D-CNN	21	0,79	0,79	0,79	0,79	0,79	1,00	0,99
RNN	21	0,79	0,78	0,79	0,78	0,78	0,89	0,89
GCN	21	0,73	0,73	0,73	0,73	0,73	0,78	0,72
Stacking	21	0,82	0,82	0,82	0,82	0,82	1,00	0,82
Deep Packet	21	0,79	0,79	0,80	0,79	0,79	0,95	0,95
FlowPic	21	0,78	0,78	0,78	0,78	0,78	0,81	0,81
FS-Net	21	0,81	0,81	0,81	0,81	0,80	0,87	0,87

Tabula 19.
Klasifikācijas veiktspēja katrā metavides datu kopas kategorijā, izmantojot 2D-CNN modeli vērtuma līmenī

Kategorija	*P.*	*R.*	*F.*
BigscreenTheatre_60FPS	0,77	0,84	0,8
BigscreenTheatre_90FPS	1	1	1
BigscreenTheatre_120FPS	1	1	1
DiRTRally2.0_60FPS	0,98	0,99	0,98
DiRTRally2.0_90FPS	0,81	0,82	0,82
DiRTRally2.0_120FPS	0,93	0,94	0,93
Hellblade_60FPS	0,81	0,87	0,84
Hellblade_90FPS	0,99	1	0,99
Hellblade_120FPS	0,94	0,9	0,92
RealityMixer+SteamVRHome_60FPS	0,84	0,93	0,88
RealityMixer+SteamVRHome_90FPS	0,54	0,55	0,54
RealityMixer+SteamVRHome_120FPS	0,89	0,85	0,87
SolarSystemAR_60FPS	0,52	0,6	0,56
SolarSystemAR_90FPS	0,96	0,92	0,94
SolarSystemAR_120FPS	0,9	0,85	0,87
TheLab_60FPS	0,66	0,63	0,64
TheLab_90FPS	0,92	0,8	0,86
TheLab_120FPS	0,82	0,8	0,81
VRChat_60FPS	0,61	0,64	0,62
VRChat_90FPS	0,88	0,74	0,8
VRChat_120FPS	1	0,99	1

Tabula 20.
Klasifikācijas veiktspējas kopsavilkums tiešraides video straumēšanas datu kopām vērtuma līmenī

Modelis	Kategorija	*P. W*	*R. W*	*F. A*	*F. I*	*A. Test*	*A. Train*	*A. Validate*
2D-CNN	78	0,„76	0,75	0,75	0,75	0,75	0,99	0,99
1D-CNN	78	0,68	0,67	0,67	0,67	0,67	0,98	0,98
RNN	78	0,68	0,65	0,65	0,65	0,65	0,71	0,71
GCN	78	0,60	0,57	0,57	0,57	0,57	0,61	0,56
Stacking	78	0,65	0,65	0,65	0,65	0,65	0,99	0,65
Deep Packet	78	0,64	0,63	0,63	0,63	0,64	0,77	0,77
FlowPic	78	0,65	0,63	0,62	0,63	0,63	0,65	0,65
FS-Net	78	0,67	0,66	0,66	0,66	0,66	0,73	0,73

Tabula 21.
Klasifikācijas veiktspēja katrā tiešraides video straumēšanas datu kopas kategorijā, izmantojot 2D-CNN modeli vērtuma līmenī

Kategorija	*P.*	*R.*	*F.*	Kategorija	*P.*	*R.*	*F.*
144P30FPS_DASH_VLS_Download	0,99	0,94	0,96	*720P60FPS_RTMP_VLS_Download*	0,55	0,63	0,59
144P30FPS_HLS_VLS_Download	1	1	1	*720P60FPS_RTMP_VLS_Upload*	0,66	0,64	0,65
144P30FPS_HTTP_FLV_VLS_Download	1	1	1	*720P60FPS_WebRTC_VLS_Download*	0,87	0,95	0,91
144P30FPS_RTMP_VLS_Download	0,99	1	1	*1080P30FPS_DASH_VLS_Download*	0,11	0,15	0,13
144P30FPS_RTMP_VLS_Upload	0,94	0,88	0,91	*1080P30FPS_HLS_VLS_Download*	0,76	0,63	0,69
144P30FPS_WebRTC_VLS_Download	1	1	1	*1080P30FPS_HTTP_FLV_VLS_Download*	0,61	0,73	0,67
240P30FPS_DASH_VLS_Download	0,96	0,9	0,93	*1080P30FPS_RTMP_VLS_Download*	0,78	0,73	0,76
240P30FPS_HLS_VLS_Download	0,99	0,96	0,98	*1080P30FPS_RTMP_VLS_Upload*	0,53	0,57	0,55
240P30FPS_HTTP_FLV_VLS_Download	0,71	0,77	0,74	*1080P30FPS_WebRTC_VLS_Download*	0,99	0,99	0,99
240P30FPS_RTMP_VLS_Download	0,91	0,83	0,87	*1080P60FPS_DASH_VLS_Download*	0,41	0,33	0,37
240P30FPS_RTMP_VLS_Upload	0,61	0,67	0,64	*1080P60FPS_HLS_VLS_Download*	0,85	0,65	0,74
240P30FPS_WebRTC_VLS_Download	0,99	1	1	*1080P60FPS_HTTP_FLV_VLS_Download*	1	0,88	0,94
360P30FPS_DASH_VLS_Download	0,92	0,81	0,86	*1080P60FPS_RTMP_VLS_Download*	1	1	1
360P30FPS_HLS_VLS_Download	0,99	0,91	0,95	*1080P60FPS_RTMP_VLS_Upload*	0,76	0,77	0,76
360P30FPS_HTTP_FLV_VLS_Download	0,52	0,59	0,55	*1080P60FPS_WebRTC_VLS_Download*	1	0,99	0,99
360P30FPS_RTMP_VLS_Download	0,52	0,61	0,56	*1440P30FPS_DASH_VLS_Download*	0,83	0,66	0,74
360P30FPS_RTMP_VLS_Upload	0,56	0,53	0,54	*1440P30FPS_HLS_VLS_Download*	0,91	0,89	0,9
360P30FPS_WebRTC_VLS_Download	0,96	0,94	0,95	*1440P30FPS_HTTP_FLV_VLS_Download*	0,94	0,99	0,96
480P30FPS_DASH_VLS_Download	0,87	0,82	0,84	*1440P30FPS_RTMP_VLS_Download*	0,18	0,34	0,24

21. tabulas turpinājums

480P30FPS_HLS_VLS_Download	0,85	0,75	0,8	*1440P30FPS_RTMP_VLS_Upload*	0,33	0,45	0,38
480P30FPS_HTTP_FLV_VLS_Download	0,86	0,91	0,89	*1440P30FPS_WebRTC_VLS_Download*	1	0,99	0,99
480P30FPS_RTMP_VLS_Download	1	0,97	0,98	*1440P60FPS_DASH_VLS_Download*	0,23	0,19	0,21
480P30FPS_RTMP_VLS_Upload	0,88	0,54	0,67	*1440P60FPS_HLS_VLS_Download*	0,68	0,69	0,69
480P30FPS_WebRTC_VLS_Download	1	1	1	*1440P60FPS_HTTP_FLV_VLS_Download*	0,17	0,17	0,17
540P30FPS_DASH_VLS_Download	0,99	1	0,99	*1440P60FPS_RTMP_VLS_Download*	0,85	0,89	0,87
540P30FPS_HLS_VLS_Download	0,85	0,71	0,78	*1440P60FPS_RTMP_VLS_Upload*	0,81	0,78	0,8
540P30FPS_HTTP_FLV_VLS_Download	0,8	0,76	0,78	*1440P60FPS_WebRTC_VLS_Download*	1	1	1
540P30FPS_RTMP_VLS_Download	0,77	0,86	0,81	*2160P30FPS_DASH_VLS_Download*	0,56	0,66	0,61
540P30FPS_RTMP_VLS_Upload	0,98	0,9	0,94	*2160P30FPS_HLS_VLS_Download*	0,72	0,62	0,67
540P30FPS_WebRTC_VLS_Download	0,92	0,89	0,9	*2160P30FPS_HTTP_FLV_VLS_Download*	0,76	0,82	0,79
720P30FPS_DASH_VLS_Download	0,65	0,99	0,79	*2160P30FPS_RTMP_VLS_Download*	0,8	0,79	0,8
720P30FPS_HLS_VLS_Download	0,37	0,45	0,41	*2160P30FPS_RTMP_VLS_Upload*	0,59	0,53	0,56
720P30FPS_HTTP_FLV_VLS_Download	1	1	1	*2160P30FPS_WebRTC_VLS_Download*	1	1	1
720P30FPS_RTMP_VLS_Download	0,99	0,99	0,99	*2160P60FPS_DASH_VLS_Download*	0,14	0,16	0.15
720P30FPS_RTMP_VLS_Upload	0,36	0,29	0,32	*2160P60FPS_HLS_VLS_Download*	0,79	0,69	0,74
720P30FPS_WebRTC_VLS_Download	0,97	0,93	0,95	*2160P60FPS_HTTP_FLV_VLS_Download*	0,19	0,16	0,17
720P60FPS_DASH_VLS_Download	0,41	0,36	0,39	*2160P60FPS_RTMP_VLS_Download*	0,11	0,1	0,11
720P60FPS_HLS_VLS_Download	0,85	0,72	0,78	*2160P60FPS_RTMP_VLS_Upload*	0,82	0,73	0,77
720P60FPS_HTTP-FLV_VLS_Download	1	0,95	0,97	*2160P60FPS_WebRTC_VLS_Download*	1	1	1

4. NOBEIGUMS

Promocijas darba kopsavilkums prezentē visaptverošu pieeju interaktīvo mediju lietotņu tīkla datu plūsmas detalizētajai klasifikācijai vairākos detalizācijas līmeņos, tostarp laika rindas, plūsmas un datu informatīvās daļas (vērtuma) līmenī.

Secinājumi, kas izriet no veikto pētījumu rezultātu analīzes

1. Tika būtiski paplašināts tradicionālās video datu plūsmas kategorijas tvērums, kas parasti aprobežojas ar vispārīgām interneta lietotņu un specifiskām protokolu datu kopām. Šajā darbā klasifikācijas tvērums tiek definēts citādi, iekļaujot trīs raksturīgas un plaši izmantotas interaktīvo mediju lietotņu kategorijas – *OTT* tiešraides straumēšanu, mākoņspēles un metavides lietotnes. Šo kategoriju kopīgas pazīmes ir, piemēram, liels datu apjoms starp klientiem un serveriem, kā arī nepārtraukta divvirzienu mijiedarbība dažādās un sarežģītās tīkla vidēs. Rezultātā veidojas datu plūsma, kurai ir raksturīgas unikālas un savstarpēji atšķirīgas raksturiezīmēs, ko netiek ņemtas vērā tradicionālajās video klasifikācijas pieejās. Atšķirībā no ierastajām klasifikācijas metodēm, kas balstās tikai vienkāršos lietotņu nosaukumos, pakalpojumu identifikatoros vai protokolu iezīmēs kā atsaucēs, šajā pētījumā tika uzsvērta un pamatota nepieciešamība pēc padziļināta konteksta izmantošanas. Tradicionālajās metodēs bieži trūkst skaidrības attiecībā uz likumsakarībām starp faktiskajām datu plūsmas īpašībām un šīs plūsmas patiesajām vērtībām. Turklāt eksistējošie pētījumi reti sniedz visaptverošu analīzi par faktoriem, kas ietekmē video datu plūsmas raksturlielumus.

Šajā darbā sistemātiski tiek pētīta ietekme uz mediju tīkla datu plūsmu, ko veido ar video saistītie faktori. Šie faktori iekļauj video kvalitātes parametrus, piemēram, izšķirtspēju, kadru skaitu sekundē, kompresijas formātu, konteineru veidu, bitu pārraides ātrumu, kustības sarežģītību, satura izcelsmi, kā arī klienta un servera aparatūras iespējas. Turklāt tiek ņemti vērā tādi pārraides parametri kā, piemēram, segmenta ilgums, *GOP* struktūra, kodeka veids, kodēšanas sākotnējie iestatījumi, maksimālais bitu pārraides ātrums, bufera atmiņas apjoms un datu plūsmas protokoli, tostarp *TCP, UDP* un *QUIC*. Pētījumā tika aplūkoti tādi lietotņu līmeņa straumēšanas protokoli kā *DASH*, *HLS, RTMP, RTCP, RTP, SRT, HTTP-FLV* un *WebRTC*. Tika

ņemti vērā tīkla stāvokļa mainīgie faktori, tādi kā, piemēram, 4G, 5G, *Wi-Fi*, pieejamais joslas platums, *QoS*, *QoE*, kā arī *VPN* vai *Tor* izmantošana datu šifrēšanai un anonimizācijai. Šie faktori tika analizēti, lai noteiktu to ietekmi uz interaktīvo mediju tīkla datu plūsmas raksturlielumiem. Pamatojoties uz iegūtajiem rezultātiem, klasifikācijas uzdevumos tika definētas datu plūsmu kategorijas, izmantojot kombinācijas no visstiprāk ietekmējošajiem faktoriem, piemēram, platformas, lietotņu nosaukuma, izšķirtspējas, kadru skaita sekunde un datu plūsmas virziena (augšupielāde vai lejupielāde). Šī pieeja ļāva izveidot ļoti detalizētas un smalki strukturēta interaktīvo mediju tīkla datu plūsmu kopas. Kopumā tika savākti vairāk nekā 84 GB datu, aptverot gan reālas lietotņu, gan ideālas testēšanas vides. Datu kopas ietver gan publiskus, gan konfidenciālus avotus, nodrošinot daudzveidīgu un reprezentatīvu interaktīvo mediju tīkla datu plūsmu nolašu apkopojumu.

2. Promocijas darbā izstrādātās detalizētās interaktīvo mediju tīkla datu plūsmu kopas raksturo ne tikai plaša kategoriju daudzveidība, bet arī vairāki klasifikācijas detalizācijas līmeņi, tostarp laika rindu līmenis, plūsmas līmenis un vērtuma līmenis. Katrs līmenis attiecas uz konkrētiem lietošanas scenārijiem. Kad *PCAP* jēldatu faili nav pieejami, laika rindu datu kopas, ko veido no pakešu skaita, baitu apjoma un laika iezīmēm, nodrošina iespēju apstrādāt datus. Šie laika apgabala (*temporal*) aspekti ļauj ātri identificēt šifrētas datu plūsmas, izmantojot ar laiku saistītus raksturlielumus bez nepieciešamības piekļūt jēldatu *PCAP* faila saturam. Plūsmas līmenī nolases atspoguļo vispārīgus datu plūsmas raksturlielumus, atpazīstot tādas specifiskās plūsmas kā "ziloņa" (lielapjoma) vai "peles" (mazapjoma) plūsmas, kas ir īpaši noderīgi dažādu interaktīvo mediju lietotņu rakstura īpatnību noteikšanai. Šīs plūsmas līmeņa īpatnības, ko apvieno ar daļēju dziļās pakešu inspekcijas (*DPI*) analīzi, veicina precīzāku lietotņu veidu un protokolu klasifikāciju. Lietderīgo datu līmenī klasifikācija ir īpaši noderīga situācijās, kad *PCAP* faila datu nolasīšanas laiks ir ierobežots vai sesiju plūsmas ir retas. Ņemot vērā interaktīvo mediju datu plūsmas "sprādzienveida" (šalts) dabu, pat īsajos datu nolasīšanas laika intervālos parasti var uzkrāt pietiekamu baitu līmeņa nolašu apjomu. Pēc šo lietderīgo datu nolašu pārveidošanas grafu formā ir iespējams izgūt gan telpiskās, gan laika apgabala pazīmes, piedāvājot pilnīgāku struktūru, salīdzinot ar vienmērīgo, viendimensiju datu formātu. Datu kopas, kas tiek veidotas visos trīs detalizācijas līmeņos, nodrošina pārskatāmu un izskaidrojamu pirmapstrādes procesu, kā arī garantē nolašu uzticamību un atkārtojamību.

3. Šajā pētījumā apskatīts un lietots konvolūcijas daudzslāņu *1D-CNN* modelis, kas bija īpaši pielāgots detalizētajai datu plūsmas klasifikācijai. Balstoties šajā arhitektūrā, tika izstrādāts modificētais *2D-CNN* modelis, kas īpaši paredzēts grafu bāzētai klasifikācijai, līdzās *RNN* un *GCN* modeļiem, kas veidoti plūsmas līmeņa apstrādei. Visos trīs detalizācijas līmeņos ieteiktais *1D-CNN* modelis pārliecinoši pārspēj esošos progresīvos modeļus, tostarp *Deep Packet*, *FlowPic* un *FS-Net*, demonstrējot augstu pielāgojamību un noturīgumu dažādos datu plūsmu scenārijos. Tika izveidots arī daudzslāņu modelis, ko veido vairāki metamācīšanās modeļi, kas izrādījās īpaši efektīvs plūsmas līmenī. Šī modeļa spēja apstrādāt augstas variācijas pazīmes un lielu nolašu skaitu uzlaboja klasifikācijas precizitāti, kas pārsniedza *1D-CNN* par vairāk nekā 10 % vienādos apstākļos. Lietderīgo datu līmenī *2D-CNN* modelis sasniedza vislabāko veiktspēju grafu bāzētajā klasifikācijā. Tā priekšrocība ir spēja izmantot divu dimensiju telpiskās un laika pazīmes, pārspējot vienas dimensijas modeļus vismaz par 10 %

kopējā klasifikācijas precizitātē. Kopumā modeļa izvēle ir būtiski atkarīga no datu kopas rakstura, pazīmēm un konkrētā scenārija prasībām. Detalizēto interaktīvo mediju tīkla datu plūsmas klasifikācijai visi izstrādātie modeļi nodrošināja augstu un stabilu veiktspēju to attiecīgajos līmeņos.

Nākotnes pētījumu virzieni

1. Laika rindu un plūsmu līmeņa klasifikācijas posmos pastāvīgs izaicinājums ir nolašu apjoma nevienmērība starp dažādām datu plūsmas kategorijām. Lai gan šo problēmu var mazināt ar lielāku neapstrādātu datu apjoma uzkrāšanu vai pasīvi vācot plūsmu datus ilgākajā laika posmā, populārs risinājums bieži vien ir izmantot sintētisko datu ģenerēšanu. Tomēr *SMOTEENN* metodes lietošana šajā darbā mazajām datu kopām bija ierobežoti iespējama, jo tā varēja apstrādāt tikai *ENN* komponenti bez jauno sintētisko nolašu ģenerēšanas. Tādēļ ir nepieciešami progresīvāki un attīstītāki datu papildināšanas paņēmieni, kas būtu vairāk piemēroti nesabalansētajam datu kopām.

2. Laika rindu līmeņa klasifikācijas uzdevumā nolašu veidošanai tika izmantoti tikai pakešu un baitu vektori, lietojot 4. algoritma universālo metodi. Turpmākajos pētījumos jāizpēta, kā citu 4. algoritma elementu lietošana un atbilstošie laika rindu sadalījumi ietekmē klasifikācijas rezultātus. Būtiski ir tas, ka ne visās kategorijās ar augstāku video izšķirtspēju vai kadru skaitu sekundē novērojams acīmredzams datu plūsmas apjoma pieaugums. Dažās kategorijās laika rindās novērojami neregulāri pakešu intensitātes uzliesmojumi un augsts trokšņu līmenis, kas negatīvi ietekmē klasifikāciju. Atbilstošu filtrēšanas metožu lietošana varētu palīdzēt mazināt šo trokšņu ietekmi.

3. Lai veiktu vērtuma līmeņa klasifikāciju, patlaban lietotā metode ar nullēm papildina pakešu datus, ja šo pakešu izmērs ir mazāks par 1250 baitiem, un saīsina tās paketes, kas pārsniedz šo 1250 baitu slieksni. Tas rada ievērojamus datu zudumus, īpaši lielajiem datu apjomiem, kas saistīti ar *TCP* tiešraides protokoliem, kuros tiek izmantotas tādas funkcijas kā *TSO*. Iespējamais uzlabojums būtu samazināt pelēktoņu grafa izmēru no 100 × 100 līdz 32 × 32, kas atbilstu 128 baitu apjomam. Lietderīgajiem datiem, kuru apjoms pārsniedz šo izmēru, varētu ģenerēt papildu grafus, lai saglabātu visu būtisko informāciju. Esošās grafu funkciju saplūšanas metodes ir parādījušas zināmu efektivitāti, uzlabojot detalizētās klasifikācijas precizitāti. Tomēr joprojām ir nepieciešamas progresīvākas saplūšanas stratēģijas, lai pilnīgāk izmantotu struktūras un konteksta informāciju no datu nolasēm.

4. Gan viendimensiju laika apgabala (*1D*) pazīmes, gan divdimensiju telpas laika apgabala (*2D*) pazīmes, kas tiek izmantotas algoritmu apmācīšanā, var būt standartizētas un saglabātas iepriekš apmācītās realizācijas formā, kas ļautu veikt interaktīvās multivides datu plūsmu tiešsaistes reāllaika klasifikāciju. Papildus klasifikācijai šīs pazīmes var būt vērtīgas arī citās jomās, piemēram, datu plūsmas prognozēšanā, anomāliju atpazīšanā un ļaunprātīgas darbības atklāšanā, ko maskē kā normālu saturu. Šiem pētniecības virzieniem ir praktiskā nozīme, kā arī ieguvumi no to turpmākās attīstības. Turklāt interaktīvās multivides tīkla datu plūsmas modeļi pēc to apmācīšanas var kalpot anonimizētas un šifrētas datu plūsmas atpazīšanai. Tas ir sevišķi svarīgi, ņemot vērā arvien plašāku ģeneratīvā mākslīgā intelekta (*AI*) lietojuma izplatību, piemēram, teksta pārveidošanu video, *GPU* dziļo mācīšanos ar ļoti augstu nolasīšanas biežumu

(*DLSS*) un kustības interpolāciju datorspēlēs un metavidē. Šīs jaunās tehnoloģijas ne tikai uzlabo lietotāju pieredzi (*QoE*), bet arī ievērojami palielina datu plūsmas apjomu. Nākotnē heterogēnajos tīklos šādi datu plūsmas modeļi var sniegt vērtīgu aprioro informāciju turpmāko pētījumu veikšanai.

BIBLIOGRĀFIJA

1. A., P. *IDC: Expect 175 Zettabytes of Data Worldwide by 2025," Network World*; 2018; Available online: https://www.networkworld.com/article/3325397/idc-expect-175-zettabytes-of-data-worldwide-by-2025.html.
2. Abbasi, M.; Shahraki, A.; Taherkordi, A. Deep Learning for Network Traffic Monitoring and Analysis (NTMA): A Survey. *Computer Communications* **2021**, *170*, 19–41, doi:10.1016/j.comcom.2021.01.021.
3. Fowdur, T. P.; Babooram, L. *Machine Learning For Network Traffic and Video Quality Analysis: Develop and Deploy Applications Using JavaScript and Node.Js*; Apress: Berkeley, CA, 2024; ISBN 979-8-8688-0353-6.
4. Nguyen, T. T. T.; Armitage, G. A Survey of Techniques for Internet Traffic Classification Using Machine Learning. *IEEE Commun. Surv. Tutorials* **2008**, *10*, 56–76, doi:10.1109/SURV.2008.080406.
5. Yan, J.; Yuan, J. A Survey of Traffic Classification in Software Defined Networks. In Proceedings of the 2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN); IEEE: Shenzhen, August 2018; pp. 200–206.
6. Wang, P.; Chen, X.; Ye, F.; Sun, Z. A Survey of Techniques for Mobile Service Encrypted Traffic Classification Using Deep Learning. *IEEE Access* **2019**, *7*, 54024–54033, doi:10.1109/ACCESS.2019.2912896.
7. Tahaei, H.; Afifi, F.; Asemi, A.; Zaki, F.; Anuar, N.B. The Rise of Traffic Classification in IoT Networks: A Survey. *Journal of Network and Computer Applications* **2020**, *154*, 102538, doi:10.1016/j.jnca.2020.102538.
8. Gunavathie; UmaMaheswariS A Survey on Traffic Prediction and Classification in SDN. In *Advances in Parallel Computing*; Jude Hemanth, D., Ambeth Kumar, V. D., Malathi, S., Eds.; IOS Press, 2020 ISBN 978-1-64368-102-3.
9. Zhao, J.; Jing, X.; Yan, Z.; Pedrycz, W. Network Traffic Classification for Data Fusion: A Survey. *Information Fusion* **2021**, *72*, 22–47, doi:10.1016/j.inffus.2021.02.009.
10. Murshed, M. G. S.; Murphy, C.; Hou, D.; Khan, N.; Ananthanarayanan, G.; Hussain, F. Machine Learning at the Network Edge: A Survey. *ACM Comput. Surv.* **2022**, *54*, 1–37, doi:10.1145/3469029.
11. Adeleke, O. A.; Bastin, N.; Gurkan, D. Network Traffic Generation: A Survey and Methodology. *ACM Comput. Surv.* **2023**, *55*, 1–23, doi:10.1145/3488375.
12. Azab, A.; Khasawneh, M.; Alrabaee, S.; Choo, K.-K. R.; Sarsour, M. Network Traffic Classification: Techniques, Datasets, and Challenges. *Digital Communications and Networks* **2024**, *10*, 676–692, doi:10.1016/j.dcan.2022.09.009.
13. Boutaba, R.; Salahuddin, M. A.; Limam, N.; Ayoubi, S.; Shahriar, N.; Estrada-Solano, F.; Caicedo, O. M. A Comprehensive Survey on Machine Learning for Networking: Evolution, Applications and Research Opportunities. *J Internet Serv Appl* **2018**, *9*, 16, doi:10.1186/s13174-018-0087-2.
14. *VNI Complete Forecast Highlights Global Internet Users: % of Population Devices and Connections per Capita Average Speeds Average Traffic per Capita per Month Global: 2021 Forecast Highlights IP Traffic*; 2016; Available online:

https://www.cisco.com/c/dam/m/en_us/solutions/service-provider/vni-forecast-highlights/pdf/Global_2021_Forecast_Highlights.pdf.
15. Sandvine *How 'App QoE' Can Increase Profitability While Improving Subscriber Satisfaction*; 2023; Available online: https://www.sandvine.com/hubfs/Sandvine_Redesign_2019/Downloads/2023/PDF/BRO-Sandvine-FEB2023.pdf.
16. Kwak, K. T.; Lee, S. Y.; Ham, M.; Lee, S. W. The Effects of Internet Proliferation on Search Engine and Over-the-Top Service Markets. *Telecommunications Policy* **2021**, *45*, 102146, doi:10.1016/j.telpol.2021.102146.
17. Che, X.; Ip, B.; Lin, L. A Survey of Current YouTube Video Characteristics. *IEEE MultiMedia* **2015**, *22*, 56–63, doi:10.1109/MMUL.2015.34.
18. Bilal, K.; Erbad, A. Impact of Multiple Video Representations in Live Streaming: A Cost, Bandwidth, and QoE Analysis. In Proceedings of the 2017 IEEE International Conference on Cloud Engineering (IC2E); IEEE: Vancouver, BC, Canada, April 2017; pp. 88–94.
19. Pan, W.; Cheng, G. QoE Assessment of Encrypted YouTube Adaptive Streaming for Energy Saving in Smart Cities. *IEEE Access* **2018**, *6*, 25142–25156, doi:10.1109/ACCESS.2018.2811416.
20. Bentaleb, A.; Taani, B.; Begen, A. C.; Timmerer, C.; Zimmermann, R. A Survey on Bitrate Adaptation Schemes for Streaming Media Over HTTP. *IEEE Commun. Surv. Tutorials* **2019**, *21*, 562–585, doi:10.1109/COMST.2018.2862938.
21. Bermudez, H.-F.; Martinez-Caro, J.-M.; Sanchez-Iborra, R.; Arciniegas, J. L.; Cano, M.-D. Live Video-Streaming Evaluation Using the ITU-T P.1203 QoE Model in LTE Networks. *Computer Networks* **2019**, *165*, 106967, doi:10.1016/j.comnet.2019.106967.
22. Jimenez, L. R.; Solera, M.; Toril, M. A Network-Layer QoE Model for YouTube Live in Wireless Networks. *IEEE Access* **2019**, *7*, 70237–70252, doi:10.1109/ACCESS.2019.2918433.
23. Lee, R.; Venieris, S. I.; Lane, N. D. Deep Neural Network–Based Enhancement for Image and Video Streaming Systems: A Survey and Future Directions. *ACM Comput. Surv.* **2022**, *54*, 1–30, doi:10.1145/3469094.
24. Sultan, M. T.; El Sayed, H. QoE-Aware Analysis and Management of Multimedia Services in 5G and Beyond Heterogeneous Networks. *IEEE Access* **2023**, *11*, 77679–77688, doi:10.1109/ACCESS.2023.3298556.
25. Zhang, Z.; Liu, L.; Lu, X.; Yan, Z.; Li, H. Encrypted Network Traffic Classification: A Data Driven Approach. In Proceedings of the 2020 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom); IEEE: Exeter, United Kingdom, October 2020; p. 7.
26. Zhao, Y.; Yang, Y.; Tian, B.; Yang, J.; Zhang, T.; Hu, N. Edge Intelligence Based Identification and Classification of Encrypted Traffic of Internet of Things. *IEEE Access* **2021**, *9*, 21895–21903, doi:10.1109/ACCESS.2021.3056216.
27. Canavese, D.; Regano, L.; Basile, C.; Ciravegna, G.; Lioy, A. Data Set and Machine Learning Models for the Classification of Network Traffic Originators. *Data in Brief* **2022**, *41*, 107968, doi:10.1016/j.dib.2022.107968.
28. Bader, O.; Lichy, A.; Dvir, A.; Dubin, R.; Hajaj, C. OSF-EIMTC: An Open-Source Framework for Standardized Encrypted Internet Traffic Classification. *Computer Communications* **2024**, *213*, 271–284, doi:10.1016/j.comcom.2023.10.011.
29. Draper-Gil, G.; Lashkari, A. H.; Mamun, M. S. I.; A. Ghorbani, A. Characterization of Encrypted and VPN Traffic Using Time-Related Features. In Proceedings of the

Proceedings of the 2nd International Conference on Information Systems Security and Privacy; SCITEPRESS - Science and and Technology Publications: Rome, Italy, 2016; pp. 407–414.
30. Habibi Lashkari, A.; Draper Gil, G.; Mamun, M. S. I.; Ghorbani, A. A. Characterization of Tor Traffic Using Time Based Features. In Proceedings of the Proceedings of the 3rd International Conference on Information Systems Security and Privacy; SCITEPRESS – Science and Technology Publications: Porto, Portugal, 2017; pp. 253–262.
31. Aceto, G.; Ciuonzo, D.; Montieri, A.; Persico, V.; Pescape, A. MIRAGE: Mobile-App Traffic Capture and Ground-Truth Creation. In Proceedings of the 2019 4th International Conference on Computing, Communications and Security (ICCCS); IEEE: Rome, Italy, October 2019; pp. 1–8.
32. Habibi Lashkari, A.; Kaur, G.; Rahali, A. DIDarknet: A Contemporary Approach to Detect and Characterize the Darknet Traffic Using Deep Image Learning. In Proceedings of the 2020 the 10th International Conference on Communication and Network Security; ACM: Tokyo Japan, November 27 2020; pp. 1–13.
33. Pham, T.-D.; Ho, T.-L.; Truong-Huu, T.; Cao, T.-D.; Truong, H.-L. MAppGraph: Mobile-App Classification on Encrypted Network Traffic Using Deep Graph Convolution Neural Networks. In Proceedings of the Annual Computer Security Applications Conference; ACM: Virtual Event USA, December 6 2021; pp. 1025–1038.
34. Dener, M.; Al, S.; Ok, G. RFSE-GRU: Data Balanced Classification Model for Mobile Encrypted Traffic in Big Data Environment. *IEEE Access* **2023**, *11*, 21831–21847, doi:10.1109/ACCESS.2023.3251745.
35. Lan, Y.; Sun, Y.; Liu, S.-P.; Ma, Z.-Z. A Real-Time Network Traffic Analysis and QoS Management Platform. In Proceedings of the 2017 IEEE 9th International Conference on Communication Software and Networks (ICCSN); IEEE: Guangzhou, May 2017; pp. 266–270.
36. Orsolic, I.; Suznjevic, M.; Skorin-Kapov, L. YouTube QoE Estimation from Encrypted Traffic: Comparison of Test Methodologies and Machine Learning Based Models. In Proceedings of the 2018 Tenth International Conference on Quality of Multimedia Experience (QoMEX); IEEE: Cagliari, May 2018; pp. 1–6.
37. Zhang, H.; Dong, L.; Gao, G.; Hu, H.; Wen, Y.; Guan, K. DeepQoE: A Multimodal Learning Framework for Video Quality of Experience (QoE) Prediction. *IEEE Trans. Multimedia* **2020**, *22*, 3210–3223, doi:10.1109/TMM.2020.2973828.
38. Wassermann, S.; Seufert, M.; Casas, P.; Gang, L.; Li, K. ViCrypt to the Rescue: Real-Time, Machine-Learning-Driven Video-QoE Monitoring for Encrypted Streaming Traffic. *IEEE Trans. Netw. Serv. Manage.* **2020**, *17*, 2007–2023, doi:10.1109/TNSM.2020.3036497.
39. Huang, Y.-F.; Lin, C.-B.; Chung, C.-M.; Chen, C.-M. Research on QoS Classification of Network Encrypted Traffic Behavior Based on Machine Learning. *Electronics* **2021**, *10*, 1376, doi:10.3390/electronics10121376.
40. Wu, Z.; Dong, Y.; Qiu, X.; Jin, J. Online Multimedia Traffic Classification from the QoS Perspective Using Deep Learning. *Computer Networks* **2022**, *204*, 108716, doi:10.1016/j.comnet.2021.108716.
41. Al-Fayoumi, M.; Al-Fawa'reh, M.; Nashwan, S. VPN and Non-VPN Network Traffic Classification Using Time-Related Features. *Computers, Materials & Continua* **2022**, *72*, 3091–3111, doi:10.32604/cmc.2022.025103.

42. Li, W.; Canini, M.; Moore, A. W.; Bolla, R. Efficient Application Identification and the Temporal and Spatial Stability of Classification Schema. *Computer Networks* **2009**, *53*, 790–809, doi:10.1016/j.comnet.2008.11.016.
43. Muehlstein, J.; Zion, Y.; Bahumi, M.; Kirshenboim, I.; Dubin, R.; Dvir, A.; Pele, O. Analyzing HTTPS Encrypted Traffic to Identify User's Operating System, Browser and Application. In Proceedings of the 2017 14th IEEE Annual Consumer Communications & Networking Conference (CCNC); IEEE: Las Vegas, NV, USA, January 2017; pp. 1–6.
44. Wei Wang; Ming Zhu; Xuewen Zeng; Xiaozhou Ye; Yiqiang Sheng Malware Traffic Classification Using Convolutional Neural Network for Representation Learning. In Proceedings of the 2017 International Conference on Information Networking (ICOIN); IEEE: Da Nang, Vietnam, 2017; pp. 712–717.
45. Arestrom, E.; Carlsson, N. Early Online Classification of Encrypted Traffic Streams Using Multi-Fractal Features. In Proceedings of the IEEE INFOCOM 2019 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS); IEEE: Paris, France, April 2019; pp. 84–89.
46. Xiao, X.; Xiao, W.; Li, R.; Luo, X.; Zheng, H.-T.; Xia, S.-T. EBSNN: Extended Byte Segment Neural Network for Network Traffic Classification. *IEEE Trans. Dependable and Secure Comput.* **2021**, 1–1, doi:10.1109/TDSC.2021.3101311.
47. Wu, H.; Wang, L.; Cheng, G.; Hu, X. Mobile Application Encryption Traffic Classification Based On TLS Flow Sequence Network. In Proceedings of the 2021 IEEE International Conference on Communications Workshops (ICC Workshops); IEEE: Montreal, QC, Canada, June 2021; pp. 1–6.
48. Shahraki, A.; Abbasi, M.; Taherkordi, A.; Kaosar, M. Internet Traffic Classification Using an Ensemble of Deep Convolutional Neural Networks. In Proceedings of the Proceedings of the 4th FlexNets Workshop on Flexible Networks Artificial Intelligence Supported Network Flexibility and Agility; ACM: Virtual Event USA, August 23 2021; pp. 38–43.
49. Grabs, E.; Chen, T.; Petersons, E.; Efrosinin, D.; Ipatovs, A.; Kluga, J.; Culkovs, D. Features Extraction for Live Streaming Video Classification with Deep and Convolutional Neural Networks. In Proceedings of the 2021 IEEE Microwave Theory and Techniques in Wireless Communications (MTTW); IEEE: Riga, Latvia, October 7 2021; pp. 58–63.
50. Ren, X.; Gu, H.; Wei, W. Tree-RNN: Tree Structural Recurrent Neural Network for Network Traffic Classification. *Expert Systems with Applications* **2021**, *167*, 114363, doi:10.1016/j.eswa.2020.114363.
51. Chen, T.; Grabs, E.; Petersons, E.; Efrosinin, D.; Ipatovs, A.; Kluga, J. Encapsulated and Anonymized Network Video Traffic Classification With Generative Models. In Proceedings of the 2022 Workshop on Microwave Theory and Techniques in Wireless Communications (MTTW); IEEE: Riga, Latvia, October 5 2022; pp. 13–18.
52. Chen, T.; Grabs, E.; Petersons, E.; Efrosinin, D.; Ipatovs, A.; Bogdanovs, N.; Rjazanovs, D. Multiclass Live Streaming Video Quality Classification Based on Convolutional Neural Networks. *Aut. Control Comp. Sci.* **2022**, *56*, 455–466, doi:10.3103/S0146411622050029.
53. Salman, O.; Elhajj, I. H.; Chehab, A.; Kayssi, A. Towards Efficient Real-Time Traffic Classifier: A Confidence Measure with Ensemble Deep Learning. *Computer Networks* **2022**, *204*, 108684, doi:10.1016/j.comnet.2021.108684.
54. Gabilondo, Á.; Fernández, Z.; Viola, R.; Martín, Á.; Zorrilla, M.; Angueira, P.; Montalbán, J. Traffic Classification for Network Slicing in Mobile Networks. *Electronics* **2022**, *11*, 1097, doi:10.3390/electronics11071097.

55. Luxemburk, J.; Čejka, T. Fine-Grained TLS Services Classification with Reject Option. *Computer Networks* **2023**, *220*, 109467, doi:10.1016/j.comnet.2022.109467.
56. Qin, T.; Cheng, G.; Wei, Y.; Yao, Z. Hier-SFL: Client-Edge-Cloud Collaborative Traffic Classification Framework Based on Hierarchical Federated Split Learning. *Future Generation Computer Systems* **2023**, *149*, 12–24, doi:10.1016/j.future.2023.07.001.
57. Chen, T.; Grabs, E.; Ipatovs, A.; Pētersons, E.; Ancāns, A. Bitrate-Based Video Traffic Classification. In Proceedings of the 2023 Photonics & Electromagnetics Research Symposium (PIERS); IEEE: Prague, Czech Republic, July 3 2023; pp. 509–514.
58. Zhu, Y.; Tao, J.; Wang, H.; Yu, L.; Luo, Y.; Qi, T.; Wang, Z.; Xu, Y. DGNN: Accurate Darknet Application Classification Adopting Attention Graph Neural Network. *IEEE Trans. Netw. Serv. Manage.* **2024**, *21*, 1660–1671, doi:10.1109/TNSM.2023.3344580.
59. Abolfathi, M.; Inturi, S.; Banaei-Kashani, F.; Jafarian, J. H. Toward Enhancing Web Privacy on HTTPS Traffic: A Novel SuperLearner Attack Model and an Efficient Defense Approach with Adversarial Examples. *Computers & Security* **2024**, *139*, 103673, doi:10.1016/j.cose.2023.103673.
60. Liu, L.; Yu, Y.; Wu, Y.; Hui, Z.; Lin, J.; Hu, J. Method for Multi-Task Learning Fusion Network Traffic Classification to Address Small Sample Labels. *Sci Rep* **2024**, *14*, 2518, doi:10.1038/s41598-024-51933-8.
61. Li, Z.; Xu, X. L2-BiTCN-CNN: Spatio-Temporal Features Fusion-Based Multi-Classification Model for Various Internet Applications Identification. *Computer Networks* **2024**, *243*, 110298, doi:10.1016/j.comnet.2024.110298.
62. Wang, C.; Zhang, W.; Hao, H.; Shi, H. Network Traffic Classification Model Based on Spatio-Temporal Feature Extraction. *Electronics* **2024**, *13*, 1236, doi:10.3390/electronics13071236.
63. Park, J.-T.; Shin, C.-Y.; Baek, U.-J.; Kim, M.-S. Fast and Accurate Multi-Task Learning for Encrypted Network Traffic Classification. *Applied Sciences* **2024**, *14*, 3073, doi:10.3390/app14073073.
64. Chen, Z.; Cheng, G.; Wei, Z.; Niu, D.; Fu, N. Classify Traffic Rather Than Flow: Versatile Multi-Flow Encrypted Traffic Classification With Flow Clustering. *IEEE Trans. Netw. Serv. Manage.* **2024**, *21*, 1446–1466, doi:10.1109/TNSM.2023.3322861.
65. Alkadi, S.; Al-Ahmadi, S.; Ben Ismail, M.M. RobEns: Robust Ensemble Adversarial Machine Learning Framework for Securing IoT Traffic. *Sensors* **2024**, *24*, 2626, doi:10.3390/s24082626.
66. Wang, C.; Finamore, A.; Michiardi, P.; Gallo, M.; Rossi, D. Data Augmentation for Traffic Classification. In *Passive and Active Measurement*; Richter, P., Bajpai, V., Carisimo, E., Eds.; Lecture Notes in Computer Science; Springer Nature Switzerland: Cham, 2024; Vol. 14537, pp. 159–186 ISBN 978-3-031-56248-8.
67. Song, Q.; Liu, J.; Zhang, C.; Wang, Z.; Wei, Z.; Ma, P. Classification of IPSec VPN Encrypted Traffic Based on a Hybrid Method. In Proceedings of the Proceedings of the 2024 2nd International Conference on Electronics, Computers and Communication Technology; ACM: Chengdu China, October 25 2024; pp. 204–209.
68. Gudla, R.; Vollala, S.; Srinivasa, K. G.; Amin, R. A Novel Approach for Classification of Tor and Non-Tor Traffic Using Efficient Feature Selection Methods. *Expert Systems with Applications* **2024**, *249*, 123544, doi:10.1016/j.eswa.2024.123544.
69. Chong, K. Spatiotemporal Influence Analysis Through Traffic Speed Pattern Analysis Using Spatial Classification. *Applied Sciences* **2024**, *15*, 196, doi:10.3390/app15010196.

70. Krupski, J.; Iwanowski, M.; Graniszewski, W. On the Right Choice of Data from Popular Datasets for Internet Traffic Classification. *Computer Communications* **2025**, *233*, 108068, doi:10.1016/j.comcom.2025.108068.
71. Xie, J.; Li, S.; Yun, X.; Si, C.; Yin, T. Sample Analysis and Multi-Label Classification for Malicious Sample Datasets. *Computer Networks* **2025**, *258*, 110999, doi:10.1016/j.comnet.2024.110999.
72. Zhan, M.; Yang, J.; Jia, D.; Fu, G. EAPT: An Encrypted Traffic Classification Model via Adversarial Pre-Trained Transformers. *Computer Networks* **2025**, *257*, 110973, doi:10.1016/j.comnet.2024.110973.
73. Li, Z.; Bu, L.; Wang, Y.; Ma, Q.; Tan, L.; Bu, F. Hierarchical Perception for Encrypted Traffic Classification via Class Incremental Learning. *Computers & Security* **2025**, *149*, 104195, doi:10.1016/j.cose.2024.104195.
74. Xu, S.-J.; Kong, K.-C.; Jin, X.-B.; Geng, G.-G. Unveiling Traffic Paths: Explainable Path Signature Feature-Based Encrypted Traffic Classification. *Computers & Security* **2025**, *150*, 104283, doi:10.1016/j.cose.2024.104283.
75. Han, Y.; Huang, Z.; Xu, P. Field-Road Trajectory Classification for Agricultural Machinery by Integrating Spatio-Temporal Clustering and Semantic Segmentation. *Computers and Electronics in Agriculture* **2025**, *233*, 110139, doi:10.1016/j.compag.2025.110139.
76. Liu, Z.; Wei, Q.; Song, Q.; Duan, C. Fine-Grained Encrypted Traffic Classification Using Dual Embedding and Graph Neural Networks. *Electronics* **2025**, *14*, 778, doi:10.3390/electronics14040778.
77. Zhu, L.; Zhang, Q.; Jian, X.; Yang, Y. Graph Convolutional Network for Traffic Incidents Duration Classification. *Engineering Applications of Artificial Intelligence* **2025**, *151*, 110570, doi:10.1016/j.engappai.2025.110570.
78. Abbasi, M.; López Flórez, S.; Shahraki, A.; Taherkordi, A.; Prieto, J.; Corchado, J.M. Class Imbalance in Network Traffic Classification: An Adaptive Weight Ensemble-of-Ensemble Learning Method. *IEEE Access* **2025**, *13*, 26171–26192, doi:10.1109/ACCESS.2025.3538170.
79. Lee, S.-H.; Lee, S.; Yun, I. Mosaic-Mixed Attention-Based Unexpected Traffic Scene Classification. *IEEE Access* **2025**, *13*, 15712–15722, doi:10.1109/ACCESS.2025.3531121.
80. Mezina, A.; Nurmi, J.; Ometov, A. Novel Hybrid UNet++ and LSTM Model for Enhanced Attack Detection and Classification in IoMT Traffic. *IEEE Access* **2025**, 1–1, doi:10.1109/ACCESS.2025.3553966.
81. Ouyang, G.; Guo, Y.; Lu, Y.; He, F. Incremental Learning for Network Traffic Classification Using Generative Adversarial Networks. *IEICE Trans. Inf. & Syst.* **2025**, *E108.D*, 124–136, doi:10.1587/transinf.2024EDP7129.
82. Wang, Y.; Song, L. Application and Optimization of Convolutional Neural Networks Based on Deep Learning in Network Traffic Classification and Anomaly Detection. *IJCAI* **2025**, *49*, doi:10.31449/inf.v49i14.7602.
83. Guan, X.; Zhu, S.; Wan, X.; Wu, Y. STARNeT: Multidimensional Spatial–Temporal Attention Recall Network for Accurate Encrypted Traffic Classification. *Journal of Network and Computer Applications* **2025**, *236*, 104109, doi:10.1016/j.jnca.2025.104109.
84. Li, Z.; Zhao, H.; Zhao, J.; Jiang, Y.; Bu, F. SAT-Net: A Staggered Attention Network Using Graph Neural Networks for Encrypted Traffic Classification. *Journal of Network and Computer Applications* **2025**, *233*, 104069, doi:10.1016/j.jnca.2024.104069.

85. Wang, Z.; Lin, W.; Chen, Y.; Vai, M. I. Robust Classification of Encrypted Network Services Using Convolutional Neural Networks Optimized by Information Bottleneck Method. *IEEE Access* **2025**, *13*, 36995–37005, doi:10.1109/ACCESS.2025.3545476.
86. Xu, S.; Han, J.; Liu, Y.; Liu, H.; Bai, Y. Few-Shot Traffic Classification Based on Autoencoder and Deep Graph Convolutional Networks. *Sci Rep* **2025**, *15*, 8995, doi:10.1038/s41598-025-94240-6.
87. Zai-jian, W.; Dong, Y.; Shi, H.; Lingyun, Y.; Pingping, T. Internet Video Traffic Classification Using QoS Features. In Proceedings of the 2016 International Conference on Computing, Networking and Communications (ICNC); IEEE: Kauai, HI, USA, February 2016; pp. 1–5.
88. Dong, Y.; Zhao, J.; Jin, J. Novel Feature Selection and Classification of Internet Video Traffic Based on a Hierarchical Scheme. *Computer Networks* **2017**, *119*, 102–111, doi:10.1016/j.comnet.2017.03.019.
89. Canovas, A.; Jimenez, J. M.; Romero, O.; Lloret, J. Multimedia Data Flow Traffic Classification Using Intelligent Models Based on Traffic Patterns. *IEEE Network* **2018**, *32*, 100–107, doi:10.1109/MNET.2018.1800121.
90. Hayashi, K.; Ooka, R.; Miyoshi, T.; Yamazaki, T. P2PTV Traffic Classification and Its Characteristic Analysis Using Machine Learning. In Proceedings of the 2019 20th Asia-Pacific Network Operations and Management Symposium (APNOMS); IEEE: Matsue, Japan, September 2019; pp. 1–6.
91. Wu, H.; Li, X.; Cheng, G.; Hu, X. Monitoring Video Resolution of Adaptive Encrypted Video Traffic Based on HTTP/2 Features. In Proceedings of the IEEE INFOCOM 2021 – IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS); IEEE: Vancouver, BC, Canada, May 10 2021; pp. 1–6.
92. Ozkan, H.; Temelli, R.; Gurbuz, O.; Koksal, O. K.; Ipekoren, A. K.; Canbal, F.; Karahan, B. D.; Kuran, M. Ş. Multimedia Traffic Classification with Mixture of Markov Components. *Ad Hoc Networks* **2021**, *121*, 102608, doi:10.1016/j.adhoc.2021.102608.
93. Wu, Z.; Dong, Y.; Jin, J.; Wei, H.-L.; Xie, G. Multimedia Traffic Classification for Imbalanced Environment. *IEEE Trans. Netw. Sci. Eng.* **2022**, *9*, 1838–1852, doi:10.1109/TNSE.2022.3153925.
94. Bukhari, S. M. A. H.; Afaq, M.; Song, W.-C. PPS: A Packets Pattern-Based Video Identification in Encrypted Network Traffic. In Proceedings of the Proceedings of the IEEE/ACM 16th International Conference on Utility and Cloud Computing; ACM: Taormina (Messina) Italy, December 4 2023; pp. 1–6.
95. Barredo Arrieta, A.; Díaz-Rodríguez, N.; Del Ser, J.; Bennetot, A.; Tabik, S.; Barbado, A.; Garcia, S.; Gil-Lopez, S.; Molina, D.; Benjamins, R.; et al. Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI. *Information Fusion* **2020**, *58*, 82–115, doi:10.1016/j.inffus.2019.12.012.
96. Khani, P.; Moeinaddini, E.; Abnavi, N. D.; Shahraki, A. Explainable Artificial Intelligence for Feature Selection in Network Traffic Classification: A Comparative Study. *Trans Emerging Tel Tech* **2024**, *35*, e4970, doi:10.1002/ett.4970.
97. Dong, H.; Lee, J. S. A. The Metaverse From a Multimedia Communications Perspective. *IEEE MultiMedia* **2022**, *29*, 123–127, doi:10.1109/MMUL.2022.3217627.
98. Carrascosa, M.; Bellalta, B. Cloud-Gaming: Analysis of Google Stadia Traffic. *Computer Communications* **2022**, *188*, 99–116, doi:10.1016/j.comcom.2022.03.006.
99. Wang, Y.; Su, Z.; Zhang, N.; Xing, R.; Liu, D.; Luan, T. H.; Shen, X. A Survey on Metaverse: Fundamentals, Security, and Privacy. *IEEE Commun. Surv. Tutorials* **2023**, *25*, 319–352, doi:10.1109/COMST.2022.3202047.

100. Martin-Perez, J.; Cominardi, L.; Bernardos, C. J.; De La Oliva, A.; Azcorra, A. Modeling Mobile Edge Computing Deployments for Low Latency Multimedia Services. *IEEE Trans. on Broadcast.* **2019**, *65*, 464–474, doi:10.1109/TBC.2019.2901406.
101. Chen, M.; Saad, W.; Yin, C.; Debbah, M. Data Correlation-Aware Resource Management in Wireless Virtual Reality (VR): An Echo State Transfer Learning Approach. *IEEE Trans. Commun.* **2019**, *67*, 4267–4280, doi:10.1109/TCOMM.2019.2900624.
102. Perfecto, C.; Elbamby, M. S.; Ser, J. D.; Bennis, M. Taming the Latency in Multi-User VR 360°: A QoE-Aware Deep Learning-Aided Multicast Framework. *IEEE Trans. Commun.* **2020**, *68*, 2491–2508, doi:10.1109/TCOMM.2020.2965527.
103. Lecci, M.; Drago, M.; Zanella, A.; Zorzi, M. An Open Framework for Analyzing and Modeling XR Network Traffic. *IEEE Access* **2021**, *9*, 129782–129795, doi:10.1109/ACCESS.2021.3113162.
104. Alencar, D.; Both, C.; Antunes, R.; Oliveira, H.; Cerqueira, E.; Rosario, D. Dynamic Microservice Allocation for Virtual Reality Distribution With QoE Support. *IEEE Trans. Netw. Serv. Manage.* **2022**, *19*, 729–740, doi:10.1109/TNSM.2021.3076922.
105. Chen, T.; Grabs, E.; Tasic, I.; Cano, M.-D. Network Traffic Analysis for eXtended Reality Applications. In Proceedings of the XVI Jornadas de Ingenieria Telematica (JITEL 2023); 2023.
106. Chiariotti, F.; Drago, M.; Testolina, P.; Lecci, M.; Zanella, A.; Zorzi, M. Temporal Characterization and Prediction of VR Traffic: A Network Slicing Use Case. *IEEE Trans. on Mobile Comput.* **2023**, 1–18, doi:10.1109/TMC.2023.3282689.
107. Balasubramanian, V.; Bouachir, O.; Al-Habashnat, A. BOLT: A Bayesian Online Learning Framework for Time Sensitive Networks in Metaverse. In Proceedings of the 2023 International Conference on Intelligent Metaverse Technologies & Applications (iMETA); IEEE: Tartu, Estonia, September 18 2023; pp. 1–7.
108. Manjunath, Y. S. K.; Szymanowski, M.; Wissborn, A.; Li, M.; Zhao, L.; Zhang, X.-P. ResLearn: Transformer-Based Residual Learning for Metaverse Network Traffic Prediction 2024.
109. Sani, Y.; Mauthe, A.; Edwards, C. Adaptive Bitrate Selection: A Survey. *IEEE Commun. Surv. Tutorials* **2017**, *19*, 2985–3014, doi:10.1109/COMST.2017.2725241.
110. Van Der Auwera, G.; David, P.; Reisslein, M. Traffic Characteristics of H.264/AVC Variable Bit Rate Video. *IEEE Commun. Mag.* **2008**, *46*, 164–174, doi:10.1109/MCOM.2008.4689260.
111. *Livestreaming Setup and Tools*; Available online: https://help.vimeo.com/hc/en-us/sections/12397317675665-Livestreaming-setup-and-tools.
112. *Live Stream on YouTube*; Available online: https://support.google.com/youtube/topic/9257891.
113. *Broadcasting Guidelines*; Available online: https://help.twitch.tv/s/article/broadcasting-guidelines.
114. *Media Transfer Guide*; Available online: https://developers.tiktok.com/doc/content-posting-api-media-transfer-guide.
115. *Requirements for Facebook Live Videos*; Available online: https://www.facebook.com/business/help/162540111070395.
116. *Content Delivery Networks*; Buyya, R., Pathan, M., Vakali, A., Eds.; Lecture Notes Electrical Engineering; Springer Berlin Heidelberg: Berlin, Heidelberg, 2008; Vol. 9; ISBN 978-3-540-77886-8.

117. Liatifis, A. L.; Tegos, S. A. T.; Mitsiou, N. A. M.; Makris, I. M.; Kyranou, K. K.; Pliatsios, D. P.; Lagkas, T. L.; Sarigiannidis, P. S. NANCY SNS JU Project “Greek In-Lab Testbed Dataset 1.”
118. Beccari, M. B.; Clavenna, A. C.; Albanese, A. A. NANCY SNS-JU Project “Italian in-Lab Testbed Dataset 1” D6.6.
119. Beccari, M. B.; Clavenna, A. C.; Albanese, A. A. NANCY SNS-JU Project “Italtel Italian in-Lab Testbed Dataset 2.”
120. Pantos, R.; May, W. *HTTP Live Streaming*; RFC Editor, 2017; p. RFC8216.
121. Reznik, Y.; Cabrera, G. Multi-Regional, Multi-CDN Delivery Optimizations Using HLS/DASH Content Steering Standard. In Proceedings of the Proceedings of the 4th Mile-High Video Conference; ACM: Denver CO USA, February 18 2025; pp. 7–12.
122. *Simple Realtime Server (SRS)*; Available online: https://github.com/ossrs/srs.
123. *Big Buck Bunny*; Available online: https://peach.blender.org.
124. Tomar, S. Converting Video Formats with FFmpeg. *Linux J.* **2006**, *2006*, 10.
125. Müller, C.; Timmerer, C. A VLC Media Player Plugin Enabling Dynamic Adaptive Streaming over HTTP. In Proceedings of the Proceedings of the 19th ACM international conference on Multimedia; ACM: Scottsdale Arizona USA, November 28 2011; pp. 723–726.
126. O’Hanlon, P.; Aslam, A. Latency Target Based Analysis of the DASH.Js Player. In Proceedings of the Proceedings of the 14th ACM Multimedia Systems Conference; ACM: Vancouver BC Canada, June 7 2023; pp. 153–160.
127. Mahmoud, H.; Abozariba, R. A Systematic Review on WebRTC for Potential Applications and Challenges beyond Audio Video Streaming. *Multimed Tools Appl* **2024**, *84*, 2909–2946, doi:10.1007/s11042-024-20448-9.
128. Chakraborty, T.; Chattopadhyay, S.; Das, S.; Kumar, U.; Senthilnath, J. Searching for Heavy-Tailed Probability Distributions for Modeling Real-World Complex Networks. *IEEE Access* **2022**, *10*, 115092–115107, doi:10.1109/ACCESS.2022.3218631.
129. Grab, E.; Petersons, E. Hurst Parameter Estimation by Wavelet Transformation and a Filter Bank for Self-Similar Traffic. *Aut. Control Comp. Sci.* **2015**, *49*, 286–292, doi:10.3103/S0146411615050041.
130. Franzl, G. Queueing Models for Multi-Service Networks. PhD Thesis, Technische Universität Wien, 2015.
131. Lin, K.; Xu, X.; Xiao, F. MFFusion: A Multi-Level Features Fusion Model for Malicious Traffic Detection Based on Deep Learning. *Computer Networks* **2022**, *202*, 108658, doi:10.1016/j.comnet.2021.108658.
132. Song, M.; Ran, J.; Li, S. Encrypted Traffic Classification Based on Text Convolution Neural Networks. In Proceedings of the 2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT); IEEE: Dalian, China, October 2019; pp. 432–436.
133. Li, F.; Ye, F. Simplifying Data Traffic Classification with Byte Importance Distillation. In Proceedings of the ICC 2021 – IEEE International Conference on Communications; IEEE: Montreal, QC, Canada, June 2021; pp. 1–6.
134. Roy, S.; Shapira, T.; Shavitt, Y. Fast and Lean Encrypted Internet Traffic Classification. *Computer Communications* **2022**, *186*, 166–173, doi:10.1016/j.comcom.2022.02.003.
135. Yang, Y.; Yan, Y.; Gao, Z.; Rui, L.; Lyu, R.; Gao, B.; Yu, P. A Network Traffic Classification Method Based on Dual-Mode Feature Extraction and Hybrid Neural Networks. *IEEE Trans. Netw. Serv. Manage.* **2023**, *20*, 4073–4084, doi:10.1109/TNSM.2023.3262246.

136. Batista, G. E. A. P. A.; Prati, R. C.; Monard, M. C. A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data. *SIGKDD Explor. Newsl.* **2004**, *6*, 20–29, doi:10.1145/1007730.1007735.
137. Lemaître, G.; Nogueira, F.; Aridas, C. K. Imbalanced-Learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. *Journal of Machine Learning Research* **2017**, *18*, 1–5.
138. Hofstede, R.; Celeda, P.; Trammell, B.; Drago, I.; Sadre, R.; Sperotto, A.; Pras, A. Flow Monitoring Explained: From Packet Capture to Data Analysis With NetFlow and IPFIX. *IEEE Commun. Surv. Tutorials* **2014**, *16*, 2037–2064, doi:10.1109/COMST.2014.2321898.
139. Brownlee, N.; Mills, C.; Ruth, G. *Traffic Flow Measurement: Architecture*; RFC Editor, 1999; p. RFC2722.
140. Yu, L.; Tao, J.; Xu, Y.; Sun, W.; Wang, Z. TLS Fingerprint for Encrypted Malicious Traffic Detection with Attributed Graph Kernel. *Computer Networks* **2024**, *247*, 110475, doi:10.1016/j.comnet.2024.110475.
141. Aouini, Z.; Pekar, A. NFStream A Flexible Network Data Analysis Framework. *Computer Networks* **2022**, *204*, 108719, doi:10.1016/j.comnet.2021.108719.
142. Chen, T.; Grabs, E.; Ipatovs, A.; Cano, M.-D. A Novel Proprietary Internet Video Traffic Dataset Generation Algorithm. *Applied Sciences* **2025**, *15*, 515, doi:10.3390/app15020515.
143. Wang, W.; Zhu, M.; Wang, J.; Zeng, X.; Yang, Z. End-to-End Encrypted Traffic Classification with One-Dimensional Convolution Neural Networks. In Proceedings of the 2017 IEEE International Conference on Intelligence and Security Informatics (ISI); IEEE: Beijing, China, July 2017; pp. 43–48.
144. Buitinck, L.; Louppe, G.; Blondel, M.; Pedregosa, F.; Mueller, A.; Grisel, O.; Niculae, V.; Prettenhofer, P.; Gramfort, A.; Grobler, J.; et al. API Design for Machine Learning Software: Experiences from the Scikit-Learn Project. **2013**, doi:10.48550/ARXIV.1309.0238.
145. Van Der Walt, S.; Schönberger, J.L.; Nunez-Iglesias, J.; Boulogne, F.; Warner, J. D.; Yager, N.; Gouillart, E.; Yu, T.; the scikit-image contributors Scikit-Image: Image Processing in Python 2014.
146. Bradski, G. The OpenCV Library. *Dr. Dobb's Journal of Software Tools* **2000**.
147. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. **2016**, doi:10.48550/ARXIV.1603.02754.
148. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library 2019.
149. Fey, M.; Lenssen, J.E. Fast Graph Representation Learning with PyTorch Geometric 2019.
150. Lotfollahi, M.; Jafari Siavoshani, M.; Shirali Hossein Zade, R.; Saberian, M. Deep Packet: A Novel Approach for Encrypted Traffic Classification Using Deep Learning. *Soft Comput* **2020**, *24*, 1999–2012, doi:10.1007/s00500-019-04030-2.
151. Shapira, T.; Shavitt, Y. FlowPic: A Generic Representation for Encrypted Traffic Classification and Applications Identification. *IEEE Trans. Netw. Serv. Manage.* **2021**, *18*, 1218–1232, doi:10.1109/TNSM.2021.3071441.
152. Liu, C.; He, L.; Xiong, G.; Cao, Z.; Li, Z. FS-Net: A Flow Sequence Network For Encrypted Traffic Classification. In Proceedings of the IEEE INFOCOM 2019 – IEEE Conference on Computer Communications; IEEE: Paris, France, April 2019; pp. 1171–1179.

153. Martinsson, P.-G.; Rokhlin, V.; Tygert, M. A Randomized Algorithm for the Decomposition of Matrices. *Applied and Computational Harmonic Analysis* **2011**, *30*, 47–68, doi:10.1016/j.acha.2010.02.003.
154. Kingma, D. P.; Ba, J. Adam: A Method for Stochastic Optimization 2014.
155. Mao, A.; Mohri, M.; Zhong, Y. Cross-Entropy Loss Functions: Theoretical Analysis and Applications 2023.
156. Dorogush, A. V.; Ershov, V.; Gulin, A. CatBoost: Gradient Boosting with Categorical Features Support 2018.
157. James, G.; Witten, D.; Hastie, T.; Tibshirani, R. *An Introduction to Statistical Learning*; Springer Texts in Statistics; Springer New York: New York, NY, 2013; Vol. 103; ISBN 978-1-4614-7137-0.

Tianhua Chen dzimis 1995. gadā Naņdzjinā (Ķīna). Rīgas Tehniskajā universitātē (RTU) ieguvis inženierzinātņu maģistra grādu telekomunikācijās (2021). Strādājis par tīkla inženieri vairākos interneta pakalpojumu sniedzēju uzņēmumos. Kopš 2023. gada ir RTU pētnieks. Zinātniskās intereses saistītas ar multimediju un paplašinātās realitātes (*XR*) lietotņu tīkla trafika analīzi, kā arī 5G *Open RAN* un *SDN* lietojumprogrammu izstrādi.