

VIDZEMES AUGSTSKOLA
INŽENIERZINĀTŅU FAKULTĀTE

**ZINĀŠANU IZGUBE UN IETVARS NO VIEDO
SISTĒMU LIETOTĀJU REAKCIJAS DATIEM**

PROMOCIJAS DARBS

Inženierzinātņu un tehnoloģiju doktora (Ph.D) zinātniskā grāda iegūšanai
elektrotehnikas, elektronikas, informācijas un komunikāciju tehnoloģiju
apakšnozarē sistēmu analīze, modelēšana un projektēšana

Autors: Mārtiņš Janševskis

Stud. apl. Nr.: SSMd18001

Darba vadītājs: Dr.sc.ing., Assoc.Prof. Kaspars Osis

Valmiera 2024

Promocijas darbs izstrādāts:

Vidzemes Augstskolas

Inženierzinātņu fakultātes

Doktora studiju programmā “Sociotehnisko sistēmu modelēšana”

no 2018.gada līdz 2024.gadam

Zinātniskais vadītājs:

Kaspars Osis

Dr.sc.ing.

Vidzemes Augstskolas asociētais profesors

OFICIĀLIE RECENZENTI

Prof. Dr.Sc.ing. **Pēteris Grabusts**

Rēzeknes Tehnoloģiju akadēmija, Latvija

Prof. Dr.sc.ing. **Jānis Grundspenķis**

Rīgas Tehniskā universitāte, Latvija

Prof. Dr. **Zeev Volkovich**

Braude College of Engineering, Izraēla

Iesniegts aizstāvēšanai Vidzemes Augstskolas Sociotehnisko sistēmu modelēšanas
promocijas padomē 2024.gadā

PATEICĪBAS

Darba autors izsaka pateicību savam zinātniskajam vadītājam ViA asociētajam profesoram Kasparam Osim par sadarbību, pacietību, izpratni, lietderīgiem komentāriem, izglītošanu un būtiskiem ierosinājumiem darba izstrādes gaitā.

Pateicos savai ģimenei – Kristīnei, Amēlijai, Ričardam, vecākiem Ivetai un Jānim, brālim un māsām par pacietību, man neesot pietiekami pieejamam vairāku gadu garumā. Pateicos par ticību un par uzmundrinājumiem.

Liels paldies arī kolēģiem uzņēmumā Monetization Solutions Inc. par iedvesmas radīšanu un atbalstu darba izstrādes procesā.

ANOTĀCIJA

Autors: Mārtiņš Janševskis, Stud.apl.nr. SSMd18001

Darba vadītājs: Dr.sc.ing., Asoc.Prof. Kaspars Osis

Zināšanu izguve un ietvars no viedo sistēmu lietotāju reakcijas datiem.

Promocijas darbs, Valmiera: Vidzemes Augstskola., 2024. - 160 lpp., 11 tabulas, 52 attēli, 15 pielikumi, bibliogrāfijas sarakstā 199 informācijas avoti.

Pasaules ekonomikas forumā Davosā 2022. gadā iekļauts jautājums par zināšanām, raksturojot situāciju sekojoši *“Ir ļoti iespējams, ka sliktam informācijas saturā, vienlaikus esam izsalkuši pēc zināšanām un tādēļ bieži vien nespējam savienot notikumus, lai paredzētu pārmaiņas, pirms tās iestājas. Katru dienu tiek izveidoti vairāk nekā četri miljardi satura vienību, sekošana līdzī visam notiekošajam ievērojami pārsniedz iespējas. Sociālo mediju organizāciju un ziņu aģentūru biznesa modeļi arvien vairāk vērsti uz to, lai cilvēkiem sniegtu vairāk to, kas viņiem patīk, tādējādi radot kameras efektu un ļaujot viegli aizmirst par kopējo ainu (Marshall & Mergenthaler, 2022).”*

Pēdējās desmitgadēs ir novērojama pāreja uz zināšanu sabiedrību, kuru raksturo tās spējas identificēt, veidot, apstrādāt, pārveidot, izplatīt un izmantot informāciju, lai veidotu un izmantotu zināšanas indivīdu attīstībai (Bindé et al., 2005). Šādā sabiedrībā intelektuālais kapitāls tiek uzskatīts par nozīmīgāko turības rādītāju, apsteidzot tādas aktīvus kā zeme, darbaspēks un finanses. Zināšanu sabiedrībai būtiskāk par informācijas radīšanu un patēriņu ir zināšanu iegūšana, pielietošana un radīšana.

Sākot ar grieķu filozofiem un beidzot ar zināšanu pārvaldības ekspertiem, ir notikuši mēģinājumi definēt zināšanas, tomēr definīcijas joprojām nav pietiekami skaidras. Bāzes zināšanu teorijas tiek integrētas divās galvenajās kategorijās: racionālismā un empīrismā. Abas teorijas atzīst, ka zināšanas ir pamatota un patiesa pārlicība (Bolisani & Bratianu, 2018). Promocijas darba ietvaros par lietderīgāko zināšanu definīciju pieņemta Ikujiro Nonaka un Hirotaka Takeuchi (Nonaka & Takeuchi, 1995) organizācijas zināšanu radīšanas teorija. Nonaka un Takeuchi izvirza apgalvojumu, ka zināšanu pamatojuma veids saistāms ar sociālo kontekstu, kurā zināšanas tiek radītas, piemēram, organizācijas kontekstu. Šāda interpretācija nozīmē, ka zināšanu rašanās vērtējama nevis ar loģiku, bet ar lietderības metriku. Uzņēmumu un organizāciju ekonomisko darbību, zināšanu izguves

efektivitāti un zināšanu sabiedrības prasību pēc nepārtrauktām inovācijām nodrošina informāciju tehnoloģijas un viedās programmatūras sistēmas.

Programmatūras sistēmas tiek veidotas no sarežģītām konstrukcijām, daļa programmatūras sistēmu īsteno to, kas jau ir uzbūvēts, un mēdz sekot zināmām un saprotamām arhitektūrām. Programmatūras izstrādes sarežģītība var strauji pieaugt, tādējādi nozīmīga ir arhitektūras, modeļu un programmatūras ietvaru izmantošana. Modeļi un ietvari nodrošina risinājumus atkārtotām problēmām, ar kurām saskaras izstrādātāji, un pielietojot ietvarus viedajās sistēmās tajās var izveidot atkārtoti lietojamās komponentes (Edwin, 2014).

Tiek pieņemts, ka viedums ir iespējams pielietojot IKT un digitālās iespējas (Garau, 2014; Kanter & Litow, 2009). Pastāv arī atšķirīgi viedokļi par vieduma jēdzienu, uzskatot, ka viedums tiek sasniegts, izmantojot vairāku faktoru, piemēram, politikas, ekonomikas, pārvaldības, izglītības, indivīdu, tehnoloģiju un ilgtspējas sadarbību, lai uzlabotu noteiktu vidi un pilnveidotu tās lietotājus (Cavada et al., 2014; Douaioui et al., 2018; Nam & Pardo, 2011). Biznesa un vadības sfērā viedumu nosaka biznesa stratēģija, ieinteresēto personu pieredze un uzvedība, kā arī tirgus apguves stratēģija (Romero et al., 2020).

Viedo sistēmu nolūks ir uztveršanas un kontroles funkcijas, lai analizētu un raksturotu noteiktu situāciju un, balstoties uz datiem, pieņemtu lēmumus prognozētā vai adaptīvā veidā, tādējādi veicot viedas darbības. Zināšanu izguvei no viedo sistēmu lietotāju reakcijas datiem ir pieaugoša tendence informācijas sistēmās, kas nodrošina iespēju sniegt kvalitatīvākus pakalpojumus sistēmu lietotājiem un atbalsta datus balstītu lēmumu pieņemšanas procesū. Zināšanu izgūvē no viedo sistēmu lietotāju reakcijas datiem ir nozīmīga sistēmu arhitektūras, modeļu un ietvaru izmantošana (Chen et al., 2018, Gokalp et al., 2016, Osman, 2019).

Darba mērķis ir izstrādāt konceptuālu zināšanu izguves ietvaru no viedo sistēmu lietotāju reakcijas datiem. Lai sasniegtu noteikto mērķi tiek analizēti pētniecības uzskati par zināšanām, viedajām sistēmām un lietotāju reakcijas datiem, tiek identificēti būtiskākie arhitektūras elementi kā arī tiek izstrādāts konceptuāls zināšanu izguves ietvars no viedo sistēmu lietotāju reakcijas datiem un uzrādīti veikta ietvara aprobācija.

Promocijas darba galvenie rezultāti ir nolasīti piecās zinātniskās konferencēs un izklāstīti septiņos zinātniskos rakstos.

ANNOTATION

Author: Martins Jansevskis, Student ID no. SSMd18001

Scientific supervisor: Dr.Sc.ing., Asoc.Prof. Kaspars Osis

Knowledge discovery and framework from user response data of intelligent systems. PhD thesis, Valmiera: Vidzeme University of Applied Sciences., 2024. - 160 pages., 11 tables, 52 pictures, 15 attachments, bibliography includes 199 sources.

The World Economic Forum in Davos in 2022 raised the issue of knowledge by describing the situation as follows: *“It could be that we are drowning in content, but starved of knowledge and therefore often fail to connect the dots to anticipate change before it becomes mainstream. With over four billion pieces of content being created each day, keeping abreast of all that is happening far exceeds our capacity to do so. Further, the business models of social media organizations and news outlets have been increasingly focused on giving people more of what they like, leading to echo chamber effects and making it easy to lose sight of the big picture (Marshall & Mergenthaler, 2022).”*

In recent decades a shift to the knowledge society has been acknowledged, characterized by its ability to identify, create, process, transform, disseminate and use information to generate and use knowledge for the development of individuals (Bindé et al., 2005). In such a society, intellectual capital is considered to be the most important indicator of wealth, ahead of assets such as land, labor and finance. The acquisition, application and creation of knowledge is more important to the knowledge society than the creation and consumption of information.

From Greek philosophers to knowledge management experts attempts have been made to define knowledge, but the definitions are still not clear. Basic knowledge theories are integrated into two main categories: rationalism and empiricism. Both theories recognize that knowledge is a valid and true belief (Bolisani & Bratianu, 2018). Ikujiro Nonaka and Hirotaka Takeuchi's (Nonaka & Takeuchi, 1995) theory of knowledge creation is accepted as the most useful definition of knowledge in the framework of the dissertation. Nonaka and Takeuchi argue that the way knowledge is based is related to the social context in which the knowledge is created, such as the organizational context. Such an interpretation means that the emergence of knowledge should be assessed not by logic but

by usability metrics. The economic performance of companies and organizations, the efficiency of knowledge retrieval and the knowledge society's demand for continuous innovation are addressed by information technology and intelligent software systems.

Software systems are made up of complex structures, some software systems implement what is already built, and tend to follow known and understandable architectures. The complexity of software development can increase rapidly, so the use of architecture, design models, and software frameworks is important. Models and frameworks provide known solutions to the recurring problems that developers face when applying them to smart systems, and may include reusable components (Edwin, 2014).

Intelligence is assumed to be possible through the use of ICT and digital opportunities (Garau, 2014; Kanter & Litow, 2009). There are also differing views on the concept of intelligence, believing that intelligence is achieved through a combination of factors such as politics, economics, governance, education, individuals, technology and sustainability to improve a certain environment and develop its users (Cavada et al., 2014; Douaioui et al., 2018; Nam & Pardo, 2011). In business and management, intelligence is determined by business strategy, stakeholder experience and behavior, and market acquisition strategy (Romero et al., 2020).

The purpose of intelligent systems is to capture and control functions in order to analyze and characterize a given situation and to make decisions based on data in a predictable or adaptive way, thus performing intelligent actions. There is a growing trend in information systems to retrieve knowledge from intelligent systems feedback, which provides an opportunity to provide better services to system users and supports a data-driven decision-making process. The use of system architecture, design models, and frameworks is important in retrieving knowledge from intelligent system user response data (Chen et al., 2018; Gokalp et al., 2016; Osman, 2019).

The aim of the work is to develop a conceptual framework of knowledge discovery from intelligent systems user response data. In order to achieve the goal research on knowledge discovery, intelligent systems and user response data are analyzed, important architectural elements are identified, as well as conceptual framework of knowledge discovery from the use of intelligent systems user response data are proposed and the approbation of the proposed framework is presented.

The main results of the dissertation have been presented in five scientific conferences and seven scientific articles.

SATURS

ANOTĀCIJA	4
ANNOTATION	6
SATURS	8
ATSLĒGVĀRDI UN SAĪSINĀJUMI	10
ATTĒLU UN TABULU APKOPOJUMS	14
IEVADS	17
1. ZINĀŠANU INTERPRETĀCIJA	24
1.1. ZINĀŠANAS.....	24
1.1.1. ZINĀŠANU IZGUVES.....	27
1.1.2. VIEDĀS SISTĒMAS.....	28
1.1.3. LIETOTĀJU REAKCIJAS DATI.....	30
1.2. SAISTOŠO PĒTĪJUMU ATLASE.....	32
1.3. ZINĀŠANU IZGUVES PROCESS.....	37
1.3.1. PIECU SOĻU MODELIS.....	40
1.3.2. ČETRU SOĻU MODELIS.....	41
1.3.3. SEŠU SOĻU MODELIS.....	43
1.3.4. STARPINDUSTRIJU STANDARTA PROCESA MODELIS (CRISP-DM).....	44
1.3.5. MODEĻU ATTĪSTĪBA.....	46
1.4. ZINĀŠANU IZGUVES IETVARI.....	48
1.4.1. KONCEPTUĀLS IETVARS INDUSTRIJAI 4.0.....	49
1.4.2. KOGNITĪVĀ SKAITĻOŠANA: ARHITEKTŪRA, TEHNOĻIJIJAS UN VIEDĀS LIETOTNES.....	52
1.4.3. INOVATĪVS LIELO DATU ANALĪZES IETVARS VIEDAJĀM PILSĒTĀM.....	54
1.4.4. IETVARU SALĪDZINĀJUMS.....	58
1.5. NODAĻAS KOPSAVILKUMS UN SECINĀJUMI.....	59
2. ZINĀŠANU IZGUVES IETVARU TEHNOĻIJIJAS APSVĒRUMI UN KRITĒRIJI	61
2.1. TEHNOĻIJIJAS APSVĒRUMI.....	62
2.1.1. DATU GLABĀŠANA.....	62
2.1.2. MĀKOŅSKAITĻOŠANAS SISTĒMAS.....	65
2.1.3. LIETOTĀJU REAKCIJU DATU PRIEKŠAPSTRĀDE.....	66
2.2. ZINĀŠANU IZGUVES UN IETEKMĒJOŠIE KRITĒRIJI.....	68
2.2.1. VISPĀRĪGĀ PERSONU DATU AIZSARDZĪBAS REGULA.....	69
2.2.2. KIBERDROŠĪBA.....	70
2.4. NODAĻAS KOPSAVILKUMS UN SECINĀJUMI.....	73
3. UIS-KDF IETVARS	75
3.1. ZINĀŠANU IZGUVES SISTĒMU IZSTRĀDES POSMI.....	76
3.1.1. PRASĪBU NOTEIKŠANA.....	76
3.1.2. TEHNOĻIJIJAS IZVĒLE.....	77

3.1.3. INFRASTRUKTŪRAS IZVEIDE.....	78
3.1.4. PĀRVALDES UN PUBLISKO LIETOTŅU IZSTRĀDE.....	79
3.1.5. MAŠĪNMĀCĪŠANĀS IESPĒJU PIELIETOŠANA.....	80
3.2. IETVARS.....	81
3.2.1. TEHNOLOĢIJU SLĀNIS.....	81
3.2.2. PUBLISKO UN PĀRVALDES LIETOTŅU SLĀŅI.....	83
3.2.3. MAŠĪNMĀCĪŠANĀS SLĀNIS.....	85
3.2.4. UIS-KDF.....	86
3.3. UIS-KDF PIELIETOJUMA METODOLOĢIJA.....	93
3.3.1. PIELIETOŠANAS IEROBEŽOJUMI.....	97
3.4. NODAĻAS KOPSAVILKUMS UN SECINĀJUMI.....	98
4. UIS-KDF APROBĀCIJA.....	100
4.1. AKTĪVĀS MĀCĪŠANĀS ATBALSTS.....	100
4.2. MOBILO LIETOTŅU NOZARE.....	105
4.3. MOBILO IEKĀRTU BATERIJU SLODZES PROGNOZES.....	113
4.3.1. DATU KOPA.....	116
4.3.2. DATU IEGUVES METODOLOĢIJA.....	118
4.3.3. STATISTISKĀ ANALĪZE.....	120
4.3.4. ALGORITMU APMĀCĪBAS REZULTĀTI.....	127
4.3.5. PAPILDU PROGNOZES PIEVIENOŠANA.....	131
4.4. NODAĻAS KOPSAVILKUMS UN SECINĀJUMI.....	137
GALVENIE REZULTĀTI UN SECINĀJUMI.....	140
IZMANTOTĀ LITERATŪRA.....	145
PIELIKUMI.....	161
I PIELIKUMS: Promocijas darba literatūras izpēte.....	162
II PIELIKUMS: IOS mobilās pakotnes koda fragments.....	201
III PIELIKUMS: Android mobilās pakotnes koda fragments.....	203
IV PIELIKUMS: Unity mobilās pakotnes koda fragments.....	206
V PIELIKUMS: API koda fragments.....	209
VI PIELIKUMS: Mašīnmācīšanās algoritma fragments.....	213
VII PIELIKUMS: Datu kopas noteikšanas fragments.....	216
VIII PIELIKUMS: Datu kopas paraugs.....	221
IX PIELIKUMS: Datu priekšapstrādes paraugs.....	222
X PIELIKUMS: Lēmumu koka vizualizācija.....	224
XI PIELIKUMS: Tehnoloģiju ceļveža paraugs.....	225
XII PIELIKUMS: Mašīnmācīšanās ietekme.....	226
XIII PIELIKUMS: Tehnoloģiju ceļvedis.....	229
XIV PIELIKUMS: Monetization Solutions Inc. atsauksme.....	231
XV PIELIKUMS: P. Kolhe atsauksme.....	232

ATSLĒGVĀRDI UN SAĪSINĀJUMI

Zināšanas - Pamatota un patiesa pārlicība (Bolisani & Bratianu, 2018a). Nonaka un Takeuchi (1995) izvirza apgalvojumu, ka zināšanu pamatojuma veids saistāms ar sociālo kontekstu, kurā zināšanas tiek radītas, piemēram, organizācijas kontekstu.

Vieda sistēma - Adaptīva un autonoma sistēma, kas spēj mijiedarboties ar vidi, pieņemot par to lēmumus, mācīties no pieredzes un rīkoties tā, lai palielinātu iespēju sasniegt izvirzītos mērķus (Luger, 2005).

Lietotāju reakcijas dati - informācija, ko sistēmu lietotāji sniedz par savu pieredzi ar noteiktu lietotni vai sistēmu (Kechagia et al., 2015).

Zināšanu izguve - process, kurā tiek iegūtas noderīgas, praktiski izmantojamas zināšanas no liela apjoma datiem, izmantojot tādas metodes kā datu ieguve, mašīnmācīšanās un statistiskā analīze (Fayyad et al., 1996).

Zināšanu izguves ietvars - sistemātiska pieeja un rīku kopums, kas var tikt pielietots, lai no liela datu apjoma izgūtu lietderīgas zināšanas. Nodrošina strukturētu pieeju, ļaujot organizācijām pieņemt datus balstītus lēmumus.

Liela apjoma dati - datu kopas, kas ir tik lielas un sarežģītas, ka to apstrādei nepieciešamas jaunas tehnoloģijas, piemēram, mākslīgais intelekts (EP, 2021).

Infrastruktūra - Darba ietvaros uz mākoņskaitļošanas pakalpojumiem balstīta infrastruktūra attiecas uz pakalpojumu, serveru, iekārtu, programmatūras un lietojumprogrammu komplektu, kas tiek izvietots mākoņservisu pakalpojumu sniedzēju serveros un kam var piekļūt, izmantojot internetu. Šī infrastruktūra ietver dažādas komponentes, kas izstrādātas, lai nodrošinātu mērogojamus un elastīgus programmatūras risinājumus bez nepieciešamības pēc lokālām iekārtām.

AI - *Artificial Intelligence* - mākslīgais intelekts.

AM - *Active Learning* - aktīvā mācīšanās.

AMK - *Active Learning Classroom* - aktīvās mācīšanās klase.

API - *Application Programming Interface* - lietojumprogrammas saskarne. Lietojumprocesos izmantojama pilna operētājsistēmas funkciju specifikācija, kā arī šo funkciju izmantošanas procedūru apraksts. Tīkla operētājsistēmās ar saskarni API tiek definēta standarta metode, kas lietojumiem nodrošina visu tīkla iespēju izmantošanu (Valsts valodas centrs et al., 2022a).

ASV - Amerikas Savienotās valstis.

BSIMM - *Building Security In Maturity Model* - programmatūras drošības nodrošināšanas ietvars, kas organizē aktivitātes, ko izmanto iniciatīvu novērtēšanai.

CI/CD - Continuous Integration (CI) / Continuous Delivery (CD) - modulāra programmatūras laidienų nepārtraukta piegāde un vienumu integrēšana.

CRISP-DM - *CRoss Industry Standard Process for Data Mining* - nozares un instrumentu neitrāls zināšanu izguves procesa modelis.

CPU - *Central Processing Unit* - centrālais procesors.

DSRM - *Design Science Research Methodology* - metodoloģiska pieeja, kas saistīta ar informācijas sistēmu komponentu izstrādi, lai komponentes nodrošinātu funkcionalitāti tās lietotājiem.

EEZ - Eiropas Ekonomiskā Zona.

EP - Eiropas Padome

ES - Eiropas Savienība.

ETL - *Extract Transform Load* - trīs posmu process, kurā dati tiek iegūti, apstrādāti un izvadīti rezultātu veidā.

G-mean - klasifikācijas precizitātes vērtību ģeometriskais vidējais.

GPU - *Graphics Processing Unit* - grafiskais procesors.

HDFS - *Hadoop Distributed File System* - liela apjoma datu glabāšanas datņu sistēma, kura atbalsta dublēšanu un mērogojamību paralēli dalītās arhitektūras sistēmās.

IaaS - *Infrastructure as a Service* - infrastruktūra kā serviss. Mākoņskaitļošanas pakalpojumu serviss, kura pakalpojumu sniedzēji nodrošina konfigurējamus skaitļošanas resursus (procesorus, datu glabātuves, tīklus un lietojumprogrammas u.c.).

IKT - Informācijas un komunikācijas tehnoloģija.

IoT - *Internet of Things* - lietu internets

KDD - *Knowledge Discovery in Databases* - zināšanu izguves procesa modelis.

LTE - *Long Term Evolution* - bezvadu platjoslas sakaru standarts mobilajām ierīcēm un datu termināļiem.

LTV - *Lifetime Value* - vērtība, kas izsaka cik vērtīgs ir kāds noteikts klients uzņēmumam visā klienta un uzņēmuma sadarbības laikā.

M2M - *Machine to Machine* - mašīnas-mašīnas komunikācija. Komunikācijas process, kurā iesaistītas tikai iekārtas.

MM - *Machine Learning* - mašīnmācīšanās.

MVP - *Minimum Viable Product* - minimālais dzīvotspējīgais produkts.

NewSQL - relāciju datu bāzes pārvaldības sistēmu klase, kuras nolūks ir nodrošināt NoSQL sistēmu mērogojamību tiešsaistes transakciju procesa apstrādei.

NoSQL - *not only SQL* - ir datu bāzes pārvaldības sistēmu klase, kuras datubāzēs var glabāt dažādus datu modeļus, tostarp atslēgas vērtību pārus, dokumentus un grafu formātus.

OAuth 2.0 - Informāciju tehnoloģijas nozares standarta protokols autorizācijas nodrošināšanai.

PaaS - *Platform as a Service* - platforma kā serviss. Mākoņskaitļošanas pakalpojumu serviss, kas nodrošina mērogojamu, sadalītu un kļūdu noturīgu mākoņpakalpojumu programmēšanas platformu datu apstrādei.

PBM - *Project Based Learning* - projektu bāzēta mācīšanās.

RAM - *Random Access Memory* - brīvpiekļuves atmiņa.

Random OverSampling - datu kopas līdzsvarošanas metode, kas ietver nejaušu piemēru atlasīšanu no mazākuma klases.

RBAC - *Role Based Access Control* - lomu bāzētas pieejas tiesības. Metode piekļuves ierobežošanai, balstoties uz lietotāju lomām uzņēmumā.

RFID - *Radio Frequency Identification* - radiofrekvenciālā identificēšana

ROC AUC - *Receiver Operating Characteristics Area Under Curve* - klasifikācijas problēmu veikspējas mērījums.

SaaS - *Software as a Service* - programmatūra kā serviss. Programmatūras lietotājiem nav nepieciešams uzturēt savu infrastruktūru vai izstrādāt specifiskus risinājumus, mākoņpakalpojumi pieejami noteiktu uzdevumu izpildei.

SMOTE-NC - *Synthetic Minority Over-sampling Technique-Nominal Continuous* - rīks sintētisku datu ģenerēšanai, lai nelīdzsvarotā datu kopā uzlabotu mazākuma mērķa klasi.

SMS - *Short Message Service* - īsziņu pakalpojums

SOC - *State of Charge* - akumulatora uzlādes līmenis attiecībā pret tā jaudu, mērvienības ir procentu punkti (0% = tukšs; 100% = pilns).

SQL - *Structured Query Language* - strukturēta vaicājumvaloda.

SSL - *Secure Sockets Layer* - drošīgzodu slānis. Protokols drošas un privātas saziņas nodrošināšanai internetā (novecojis).

TLS - *Transport Layer Security* - ir kriptogrāfijas protokols, kas paredzēts sakaru drošības nodrošināšanai internetā, aizstāj novecojušo SSL protokolu.

UAC - *User Acquisition Cost* - lietotāja ieguves izmaksas. Viena jauna organizācijas lietotāja piesaistīšanas kopējās izmaksas.

UI - *User Interface* - lietotāja saskarne.

UIS-KDF - *User Interaction and Response-based Knowledge Discovery Framework* - Konceptuāls zināšanu izguves ietvars no viedo sistēmu lietotāju reakcijas datiem.

USD - *United States dollar* - Amerikas Savienoto valstu dolārs.

VDAR - Vispārīgā Datu Aizsardzības Regula.

VM - *Virtual Machine* - virtuālā mašīna. Virtuāla datu apstrādes sistēma, kas šķietami nodota katra atsevišķa lietotāja rīcībā, bet kuras darbība tiek nodrošināta, virtuālo mašīnu lietotājiem kopīgi izmantojot reālās datu apstrādes sistēmas resursus (Valsts valodas centrs et al., 2022b).

4G - ceturtās paaudzes mobilo sakaru tīkls, kas izmanto interneta tehnoloģiju LTE. Internets 4G tīklā ir vairākas reizes ātrāks par iepriekšējās versijas potenciāli iespējamo, kā arī ar uzlabotu stabilitāti un mazāku reakcijas laiku (LMT, 2022).

5G - piektās paaudzes mobilo sakaru tīkls.

ATTĒLU UN TABULU APKOPOJUMS

Attēli:

- 1.1. attēls. Viedo sistēmu meta modelis.
- 1.2. attēls. Lietotāju reakcijas dati.
- 1.3. attēls. Zināšanu izguves procesa modelis.
- 1.4. attēls. Zināšanu izguves cikls, piecu soļu modelis.
- 1.5. attēls. Zināšanu izguves cikls, četru soļu modelis.
- 1.6. attēls. Zināšanu izguves cikls, sešu soļu modelis.
- 1.7. attēls. CRISP-DM procesa modelis.
- 1.8. attēls. Zināmākie zināšanu izguves modeļi.
- 1.9. attēls. Gokalp et al., piedāvātais konceptuālais ietvars.
- 1.10. attēls. Programmēšanas mezgli Gokalp et al. ietvarā.
- 1.11. attēls. Kognitīvās skaitļošanas attīstība.
- 1.12. attēls. Kognitīvās skaitļošanas ietvara arhitektūra.
- 1.13. attēls. SCDAP ietvara arhitektūra.
- 1.14. attēls. SCDAP ietvara aprobācija ar Apache Hadoop.
- 2.1. attēls. Lielo datu servisa arhitektūra.
- 2.2. attēls. ETL process.
- 2.3. attēls. NoSQL datubāzu tipi.
- 2.4. attēls. Mākoņskaitļošanas pakalpojumu arhitektūra.
- 2.5. attēls. Spark un Storm salīdzinājums.
- 2.6. attēls. Datu drošība.
- 2.7. attēls. Zināšanu izguves sistēmu ierobežojošie kritēriji
- 3.1. attēls. Zināšanu izguves sistēmas izstrādes process.
- 3.2. attēls. Zināšanu izguves ietvara tehnoloģiju slānis.
- 3.3. attēls. Publisko un pārvaldes lietotņu slāņi.
- 3.4. attēls. MM slānis.
- 3.5. attēls. UIS-KDF ietvars.
- 3.6. attēls. UIS-KDF ietvara pielietošana.
- 4.1. attēls. Adaptēts UIS-KDF ietvars zināšanu izguvei no viedo sistēmu lietotāju pieredzes datiem aktīvās mācīšanās atbalstam.
- 4.2. attēls. Mobilās pakotnes produktu skats, pielāgots noteiktai spēlei.

- 4.3. attēls. Adaptētais UIS-KDF ietvars.
- 4.4. attēls. UIS-KDF ietvara mašīnmācīšanās slāņa viena algoritma apmācības process.
- 4.5. attēls. Android iekārtas izsekošanas datņu apjoms.
- 4.6. attēls. Mainīgo aprakstošā statistika.
- 4.7. attēls. Spirmena korelācija starp GPU slodzi un akumulatora uzlādes stāvokli.
- 4.8. attēls. Vienkāršā korelāciju analīze.
- 4.9. attēls. Neparametriskā Spirmena korelācija starp CPU frekvenci un akumulatora uzlādes stāvokli.
- 4.10. attēls. Neparametriskā Spirmena korelācija starp RAM un akumulatora uzlādes stāvokli.
- 4.11. attēls. Neparametriskā Spirmena korelācija starp akumulatora temperatūru un akumulatora uzlādes stāvokli.
- 4.12. attēls. Uzlādes stāvokļa ierakstu dalījums atbilstoši testēšanas scenārijiem.
- 4.13. attēls. Akumulatora temperatūras ierakstu dalījums atbilstoši testēšanas scenārijiem.
- 4.14. attēls. GPU frekvences ierakstu dalījums atbilstoši testēšanas scenārijiem.
- 4.15. attēls. Korelāciju grafiks.
- 4.16. attēls. GPU frekvences mainīgā vērtības.
- 4.17. attēls. CPU frekvences mainīgā vērtības.
- 4.18. attēls. Dummy klasifikatora rezultātu atskaite.
- 4.19. attēls. Dummy klasifikatora rezultātu matrica.
- 4.20. attēls. Random Forest klasifikatora rezultātu atskaite.
- 4.21. attēls. Random Forest klasifikatora rezultātu matrica.
- 4.22. attēls. Dumpsys UI atgrieztie dati.
- 4.23. attēls. Apkopotās renderēto kadru kopas paraugs.
- 4.24. attēls. Korelāciju analīze kadru statistikai un akumulatora patēriņam.
- 4.25. attēls. Datu kopas mainīgo ietekme.

Tabulas

- 1.1. tabula. Meklēšanas scenāriju rezultāti
- 1.2. tabula. Literatūras izpētes, rezultātu apkopojums
- 1.3. tabula. Ietvaru salīdzinājums
- 2.1. tabula. Zināšanu izguves ietvaru izstrādes tehnoloģiskie kritēriji
- 3.1. tabula. UIS-KDF ietvara pazīmes un nepilnības
- 3.2. tabula. UIS-KDF ietvara kvantitatīvo rādītāju novērtējuma pieeja

4.1. tabula. Lietotāju reakcijas datu potenciālās kopas

4.2. tabula. Pētījumam noteiktā datu kopa

4.3. tabula. UIS-KDF aprobācijas izvērtējuma apkopojums

4.4. tabula. UIS-KDF viena modeļa apmācības izvērtējuma apkopojums

4.5. tabula. UIS-KDF ietvara aprobācija tehnoloģiju ietvaros

IEVADS

Pēdējās desmitgadēs ir novērojama pāreja uz zināšanu sabiedrību. Zināšanu sabiedrību raksturo tās spējas identificēt, veidot, apstrādāt, pārveidot, izplatīt un izmantot informāciju, lai veidotu un izmantotu zināšanas indivīdu attīstībai (Bindé et al., 2005). Šādā sabiedrībā intelektuālais kapitāls tiek uzskatīts par nozīmīgāko turības rādītāju, apsteidzot tādus aktīvus kā zeme, darbaspēks un finanses. Zināšanu sabiedrības nozīmīgākās iezīmes var raksturot sekojoši: masveida zināšanu ražošana, pārraide un pielietošana; preču un pakalpojumu vērtību nosaka to izstrādei nepieciešamās zināšanas; lielākajai daļai iedzīvotāju ir pieejamas informācijas un komunikācijas tehnoloģijas un internets; liela daļa darbaspēka ir zināšanu darbinieki; tiek investēti apjomīgi līdzekļi izglītībā, pētniecībā un attīstībā; organizācijām nepieciešams nepārtraukti ieviest inovācijas (Knowledge Society, 2022). Pretēji zināšanu sabiedrībai, informācijas laikmetu raksturojošas iezīmes ir informācijas radīšana un patēriņš, nestandarta pieeju izmantošana darba procesā un nestandarta informācijas apstrāde un izmantošana (Osis, 2011). Zināšanu sabiedrībai būtiskāk par informācijas radīšanu un patēriņu ir zināšanu iegūšana, pielietošana un radīšana.

Sākot ar grieķu filozofiem un beidzot ar zināšanu pārvaldības ekspertiem, ir notikuši mēģinājumi definēt zināšanas, tomēr definīcijas joprojām nav pietiekami skaidras. Bāzes zināšanu teorijas tiek integrētas divās galvenajās kategorijās: racionālismā un empīrismā. Abas teorijas atzīst, ka zināšanas ir pamatota un patiesa pārlicība (Bolisani & Bratianu, 2018). Promocijas darba ietvaros par lietderīgāko zināšanu definīciju pieņemta Ikujiro Nonaka un Hirotaka Takeuchi (Nonaka & Takeuchi, 1995) organizācijas zināšanu radīšanas teorija. Nonaka un Takeuchi izvirza apgalvojumu, ka zināšanu pamatojuma veids saistāms ar sociālo kontekstu, kurā zināšanas tiek radītas, piemēram, organizācijas kontekstu. Šāda interpretācija nozīmē, ka zināšanu rašanās vērtējama nevis ar loģiku, bet ar lietderības metriku. Organizācijas zināšanu teorija nosaka praktiskus un izmērāmus pamatojuma kritērijus, piemēram, izmaksas, peļņu un apmēru kādā produkts vai pakalpojums var veicināt uzņēmuma ekonomisko darbību (Nonaka & Takeuchi, 1995). Uzņēmumu un organizāciju ekonomisko darbību, zināšanu izguves efektivitāti un zināšanu sabiedrības prasību pēc nepārtrauktām inovācijām nodrošina informāciju tehnoloģijas un viedās programmatūras sistēmas.

Tiek pieņemts, ka viedums ir iespējams pielietojot IKT un digitālās iespējas (Garau, 2014; Kanter & Litow, 2009). Pastāv arī atšķirīgi viedokļi par vieduma jēdzienu, uzskatot,

ka viedums tiek sasniegts, izmantojot vairāku faktoru, piemēram, politikas, ekonomikas, pārvaldības, izglītības, indivīdu, tehnoloģiju un ilgtspējas sadarbību, lai uzlabotu noteiktu vidi un pilnveidotu tās lietotājus (Cavada et al., 2014; Douaioui et al., 2018; Nam & Pardo, 2011). Biznesa un vadības sfērā viedumu nosaka biznesa stratēģija, ieinteresēto personu pieredze un uzvedība, kā arī tirgus apguves stratēģija (Romero et al., 2020). Plašāk lietotā sistēmas definīcija: “sistēma ir savstarpēji savienotu daļu kopums ar īpašībām, kas ir lielākas par šo daļu īpašību summu. Sistēmai ir robežas un atribūtu kopums, kas padara to unikālu salīdzinājumā ar citām sistēmām (Naudet et al., 2009)”. Viedās un programmatūras sistēmas darbības procesos uzkrāj lietotāju reakcijas datus, kurus var pielietot zināšanu izguvei.

Lietotāju reakcijas dati ir informācija, ko sistēmu lietotāji sniedz par savu pieredzi ar noteiktu lietotni vai sistēmu. Lietotāju reakcijas datiem var būt divas kategorijas - aktīvie uzkrātie (atsauksmes, komentāri, intervijas, u.c.) un pasīvi uzkrātie. Pasīvi uzkrātie lietotāju reakcijas dati ir mijiedarbības dati ar sistēmu, piemēram, veiktās darbības sistēmā, veiktās izvēles, pavadītais laiks u.c. (Kechagia, Mitropoulos & Spinellis, 2015). Pasīvos lietotāju reakcijas datus iespējams ievākt par visiem sistēmas lietotājiem, iekļaujot tos lietotājus, kuri nepiedalās aktīvi uzkrāto datu nodrošinošās darbībās. Programmatūras sistēmu pasīvie lietotāju reakcijas dati var nodrošināt efektīvāku un datus balstītu lēmumu pieņemšanas procesu, piemēram, hipotēžu validāciju, lai noteiktu kāda izskata komponentes darbojas labāk vai kādi kvalitātes kritēriji ir būtiski viedo programmatūras sistēmu lietotājiem (Dugar, 2019).

Programmatūras sistēmas tiek veidotas no sarežģītām konstrukcijām, daļa programmatūras sistēmu īsteno to, kas jau ir uzbūvēts, un mēdz sekot zināmām un saprotamām arhitektūrām. Programmatūras izstrādes sarežģītība var strauji pieaugt, tādējādi nozīmīga ir arhitektūras, modeļu un programmatūras ietvaru izmantošana. Modeļi un ietvari nodrošina zināmus risinājumus atkārtotām problēmām, ar kurām saskaras izstrādātāji, tos pielietojot viedajās sistēmās, var iekļaut atkārtoti lietojamas komponentes (Edwin, 2014).

Viedo sistēmu nolūks ir uztveršanas un kontroles funkcijas, lai analizētu un raksturotu noteiktu situāciju un, balstoties uz datiem, pieņemtu lēmumus prognozētā vai adaptīvā veidā, tādējādi veicot viedas darbības. Viedajās sistēmās pielietotās informāciju un komunikāciju tehnoloģijas, un lietojumprogrammas nodrošina lielus datu apjomus. Zināšanu izguvei no viedo sistēmu lietotāju reakcijas datiem ir pieaugoša tendence informācijas sistēmās, kas nodrošina iespēju sniegt kvalitatīvākus pakalpojumus sistēmu

lietotājiem un atbalsta datus balstītu lēmumu pieņemšanas procesu. Lai izgūtu lietderīgas zināšanas no viedo sistēmu lietotāju reakcijas datiem, uzkrātās datu kopas no dažādiem sistēmas domēniem nepieciešams integrēt un analizēt. Zināšanu izgūvē no viedo sistēmu lietotāju reakcijas datiem ir nozīmīga sistēmu arhitektūras, modeļu un ietvaru izmantošana (Chen et al., 2018, Gokalp et al., 2016, Osman, 2019).

Promocijas darba mērķis

Darba mērķis ir izstrādāt konceptuālu ietvaru (UIS-KDF) zināšanu izgūvei no viedo sistēmu lietotāju reakcijas datiem, atbalstot elastīgu un mērogojamu intelektuālu sistēmu ar iegultām zināšanu izgūves spējām izstrādi.

Darba uzdevumi

Promocijas darba mērķa sasniegšanai ir izvirzīti šādi uzdevumi:

- veikt pētniecības uzskatu apkopošanu un analīzi attiecībā uz tādiem traktātiem kā zināšanas, viedās sistēmas un lietotāju reakcijas dati;
- identificēt un analizēt pieejamos zināšanu izgūves ietvarus un to tehnoloģiju aprobāciju;
- identificēt būtiskos zināšanu izgūves ietvaru arhitektūras elementus un ietekmējošos kritērijus;
- izstrādāt ietvara novērtēšanas pieeju un apkopot rezultātus;
- izstrādāt konceptuālu zināšanu izgūves ietvaru no viedo sistēmu lietotāju reakcijas datiem (UIS-KDF);
- izstrādāt UIS-KDF ietvara aprobācijas un pielietojuma paraugus.

Pētījuma objekts

Informācijas sistēmu analīze, modelēšana un projektēšana elastīgu un mērogojamu intelektuālu sistēmu ar zināšanu izgūšanas spēju izstrādei.

Pētījuma priekšmets

Promocijas darba pētījuma priekšmets ir UIS-KDF zināšanu izgūves ietvara no viedo sistēmu lietotāju reakcijas datiem izstrāde.

Pētījuma jautājums

Kāds mūsdienīgs tehnoloģiskais risinājums nodrošina elastīgu un mērogojamu intelektuālu sistēmu ar zināšanu izgūšanas spēju izstrādes procesu?

Pētījuma metodes

Promocijas darbā pielietotas sekojošas pētījumu metodes:

- Literatūras analīze un tehnoloģisko iespēju apkopošana UIS-KDF izstrādei;
- Datu iegūves metodes - pasīvi uzkrājamo lietotāju reakcijas datu ievākšana;

- Statistiskās analīzes metodes, datu priekšapstrāde un analīze;
- Kvalitatīvās metodes - speciālistu konsultācijas;
- Eksperimenti - UIS-KDF ietvara aprobācija.

Novitāte

- izstrādāta pieeja literatūras avotu atlasei par zināšanu izguves sistēmām, lietotāju reakcijas datiem un viedajām sistēmām, izveidojot aptverošu avotu sarakstu un saistošo pētījumu apkopojumu;
- izpētīti un salīdzināti zināšanu izguves ietvari un identificēti to trūkumi papildinot zinātnisko literatūru par zināšanu izguves ietvariem;
- izstrādāts konceptuāls zināšanu izguves ietvars no viedo sistēmu lietotāju reakcijas datiem UIS-KDF, kas balstīts uz promocijas darbā apskatīto zināšanu izguves procesu;
- izstrādāta UIS-KDF ietvara pielietojuma kvantitatīvo rādītāju novērtējuma pieeja, kas pielāgojama arī citu zināšanu izguves ietvaru novērtējumam;
- izstrādāta UIS-KDF pielietojuma metodoloģija;
- izveidoti divi mašīnmācīšanās risinājumi pielietošanai mobilo lietotņu nozarē, kuru nolūks ir prognozēt vai noteikta iekārta var tikt monetizēta un prognozētu mobilās iekārtas akumulatora enerģijas patēriņu dažādos lietošanas scenārijos;
- izstrādāti UIS-KDF ietvara aprobācijas piemēri, papildinot zinātnisko literatūru par zināšanu izguves sistēmu izstrādi un izmantošanu.

Pētījumā izvirzītās tēzes

- Esošo zināšanu izguves sistēmu izstrādes ietvaru skaits ir ierobežots, un tie ir pielāgoti konkrētām jomām, kas apgrūtina to pielietošanu dažādās nozarēs;
- UIS-KDF ietvara izmantošana zināšanu izguves sistēmu plānošanas sākuma posmā nodrošina strukturētu vadlīniju arhitektūras izstrādei un organizatoriskajai plānošanai.

Pētījuma praktiskā nozīmība

Promocijas darba praktiskā nozīmība saistās ar izstrādāto adaptējamo zināšanu izguves ietvaru no viedo sistēmu lietotāju reakcijas datiem, kas pielieto mākoņskaitļošanas tehnoloģijas, kā arī konkrētas sistēmas realizāciju jaunuzņēmuma organizācijas mērķu atbalstam.

Darba aprobācija

Zinātniskās publikācijas starptautiski citējamās datubāzēs iekļautajos izdevumos:

- Jansevskis, M., Osis, K. (2024). Securing the Future: The Role of Knowledge Discovery Frameworks. In: Sipola, T., Alatalo, J., Wolfmayr, M., Kokkonen, T. (eds) Artificial Intelligence for Security. Springer, Cham. https://doi.org/10.1007/978-3-031-57452-8_5
- Jansevskis, M., Osis, K. (2024). User Interaction and Response-Based Knowledge Discovery Framework. Information and Software Technologies. 29th International Conference, ICIST 2023, Kaunas, Lithuania, October 12–14, 2023, Proceedings. Springer. https://doi.org/10.1007/978-3-031-48981-5_8 [Scopus datu bāzē]
- Jansevskis, M., Osis, K. (2023). Knowledge Discovery Frameworks and Characteristics. Baltic Journal of Modern Computing, 11(4). <https://doi.org/10.22364/bjmc.2023.11.4.08> [Scopus datu bāzē]
- Jansevskis, M., Osis, K. (2022). State of Knowledge Discovery Process Models and Frameworks. Society. Technology. Solutions. Proceedings Of The International Scientific Conference, 2, 14, DOI: <https://doi.org/10.35363/ViA.sts.2022.81>
- Jansevskis, M., & Osis, K. (2020). Framework and Knowledge Discovery in Learning Feedback Data Based Active Learning Ecosystem For Teaching Support. International Conference of Education, Research and Innovation. 5700–5708. DOI: <https://doi.org/10.21125/iceri.2020.1226>
- Jansevskis, M., Osis, K. (2020). Knowledge Discovery and Framework for Purchase Behavior Analysis in Mobile Gaming Applications. International Conference Computer Graphics, Visualization, Computer Vision and Image Processing, 247–251, ISBN: 978-989-8704-21-4 [Scopus datu bāzē]
- Jansevskis, M., & Osis, K. (2018). Machine Learning and on 5G Based Technologies Create New Opportunities to Gain Knowledge. 2018 2nd European Conference on Electrical Engineering and Computer Science (EECS), 376–381, DOI: <https://doi.org/10.1109/EECS.2018.00076> [Scopus datu bāzē]

Konferences:

- Information and Software Technologies. 29th International Conference, ICIST 2023, Kaunas, Lithuania, October 12–14, 2023. Publikācija - User

Interaction and Response-Based Knowledge Discovery Framework. Information and Software Technologies;

- International Scientific Conference SOCIETY. TECHNOLOGY. SOLUTIONS. Valmiera, Latvija, April 8, 2022. Publikācija - State of Knowledge Discovery Process Models and Frameworks;
- International Conference of Education, Research and Innovation. Seville, Spain. 9-10 November, 2020, Publikācija - Framework and Knowledge Discovery in Learning Feedback Data Based Active Learning Ecosystem For Teaching Support;
- International Conference Computer Graphics, Visualization, Computer Vision and Image Processing. Zagreb, Croatia, July 23 – 25 2020, Publikācija - Knowledge Discovery and Framework for Purchase Behavior Analysis in Mobile Gaming Applications;
- European Conference on Electrical Engineering and Computer Science (EECS), Bern, Switzerland, December 20-22. 2018, Publikācija - Machine Learning and on 5G Based Technologies Create New Opportunities to Gain Knowledge.

Pētniecības projekti:

- Projekts "Stiprināt augstākās izglītības institūciju akadēmisko personālu stratēģiskās specializācijas jomās (Development of human resources and the academic personnel at Vidzeme University of Applied Sciences)", ID 8.2.2.0/18/A/012 (SAM822);
- Projekts Eiropas nākamās paaudzes mazās pilsētas (Next Generation Micro Cities of Europe), Nr. UIA03-250;
- Valmieras pilsētas pašvaldības finansēts projekts "5G sniegto sociotehnisko iespēju analīze un piedāvājuma izstrāde viedās pilsētvides un pētījumu projektu attīstība";
- European Smart Growth Operational programma 2014-2020 ar kapitāla fondu līdzdalību apakšgrupu 1.3.1: Pētniecības un attīstības projektu atbalsts konceptuālu projektu izstrādei - Bridge Alfa.

Darba struktūra

Promocijas darbs sastāv no ievada, četrām daļām, secinājumiem, bibliogrāfijas un pielikumiem.

Darba ievadā uzrādīta veikto pētījumu aktualitāte, definēts darba mērķis, pētījuma objekts, pētījuma priekšmets, pētījuma jautājums, tēzes un uzdevumi, uzskaitītas promocijas darba izstrādē lietotās metodes, uzrādīta pētījumu zinātniskā novitāte un rezultātu praktiskā nozīmība, kā arī ir raksturota darba aprobācija.

Promocijas darba pirmajā daļā veikta pētniecības terminu par zināšanu interpretāciju apkopošana. Tiek definēta saistošo pētījumu atlases metodoloģija, apskatīti zināšanu izguves procesu modeļi un aprakstīti zināšanu izguves ietvari.

Promocijas darba otrajā daļā tiek apskatīti zināšanu izguves ietvaru tehnoloģiskie apsvērumi un ietekmējošie faktori.

Promocijas darba trešā daļa veltīta UIS-KDF ietvara izstrādei, iekļaujot tehnoloģiskos aspektus un sniedzot aprakstu pārejai no UIS-KDF ietvara uz tehnoloģisku ietvaru.

Promocijas darba ceturtajā daļā tiek sniegtas UIS-KDF ietvara trīs aprobācijas pieejas. Darba rezultāti un secinājumi ir apkopoti promocijas darba noslēguma sadaļā.

1. ZINĀŠANU INTERPRETĀCIJA

1.1. ZINĀŠANAS

Zināšanu traktāts raisījis diskusijas par jēdziena abstraktumu un saikni ar fizisko pasauli. Sākot ar grieķu filozofiem un beidzot ar zināšanu pārvaldības ekspertiem, ir notikuši mēģinājumi definēt zināšanas, taču esošās definīcijas joprojām nav pietiekami skaidras (Bolisani & Bratianu, 2018b). Filozofi, sākot ar Platonu un Aristoteli, izstrādāja Epistemoloģiju kā zināšanu teoriju, mēģinot atbildēt uz pamatjautājumu: kas ir zināšanas? Zināšanu definēšana un to būtības izskaidrošana ir izrādījusies netverama bez pārliecinoša un vispārpieņemta rezultāta (Neta & Pritchard, 2009). Lielākā daļa teoriju ir integrētas divās galvenajās perspektīvās: racionālismā un empīrismā. Abas teorijas atzīst, ka zināšanas ir pamatota patiesa pārliecība (Bolisani & Bratianu, 2018b).

Racionālisms pieņem, ka zināšanas ir pamatojuma process un ka indivīda maņu pieredzei nav nozīmes. Zināšanas var iegūt tikai ar racionālu argumentāciju, kas pamatota ar aksiomām, un tās ir jānošķir no viedokļa, kas ir indivīda maņu produkts (Bolisani & Bratianu, 2018b). Racionālisms ir saistīts ar fundamentālismu, viedokli, ka indivīds zina dažas patiesības, nepamatojot pārliecību par citām patiesībām un ka pēc tam šīs pamatzināšanas tiek izmantotas, lai uzzinātu vairāk patiesību (Markie & Zlata, 2017).

Empīrisms izveidojās kā racionālismam pretēja perspektīva. Tiek pieņemts, ka idejas un formas nevar atdalīt no fiziskiem objektiem un maņu informācijas. Zināšanas netiek radītas, un tās nav iedzimtas noteiktā formā. Zināšanas tiek izveidotas, izmantojot indivīda maņu saskarni ar reālo pasauli, un indivīdu prāts to apstrādā (Bolisani & Bratianu, 2018b). Empīrisms noliedz iedzimtu zināšanu koncepciju, pieņemumu, ka indivīdiem ir iedzimtas idejas. Empīrismā sajūtas ir vienīgais ideju avots un tā kā saprāts nedod nekādas zināšanas, tas noteikti nedod pilnīgākas zināšanas (Markie & Zalta, 2017).

Darba ietvaros lietderīgāka ir Ikujiro Nonaka un Hirotaka Takeuchi (Nonaka & Takeuchi, 1995b) organizācijas zināšanu radīšanas teorija. Ikujiro Nonaka un Hirotaka Takeuchi uzskata, ka lietderīgāks zināšanu pamatojuma veids jāsaista ar sociālo kontekstu, kurā zināšanas tiek radītas, piemēram, organizācijas kontekstu. Praksē jaunu zināšanu rašanās jāvērtē nevis ar loģiku, bet ar lietderības metriku. Nonaka un Takeuchi norāda praktiskus pamatojuma kritērijus, piemēram, izmaksas, peļņu un apmēru, kādā produkts var veicināt uzņēmuma ekonomisko darbību. Augstākie vadītāji prasa zināšanu saskaņotību ar organizācijas stratēģisko redzējumu, savukārt vidējā līmeņa vadītāji meklē praktisku pielietojumu (Nonaka & Takeuchi, 1995b).

Zināšanas, balstoties uz Dombrowski, tiek iedalītas trīs veidos (Dombrowski et al., 2013):

- pieredzes zināšanas. Tās zināšanas, kuras indivīdi iegūst no tiešas saiknes ar vidi, izmantojot maņu orgānus. Piemēram, zināšanas par to, kas ir sniegs, lietus, saulesgaisma, runa, tauste, skaņas, u.tml.;
- prasmes. Zināšanas par to kā kaut ko darīt. Balstītas ar pieredzes zināšanām, tās ir labi strukturētas un uz darbību orientētas, kuras indivīdi iegūst, atkārtoti veicot noteiktu uzdevumu un mācoties to izpildīt. Prasmes ir veids, kā iemācīties peldēt, braukt ar velosipēdu, slēpot, spēlēt klavieres vai veikt citas līdzīgas aktivitātes;
- apgalvojumi par zināšanām. Tās zināšanas, kuras indivīdi zina, vai arī domā, ka zina. Šeit iekļaujamas arī izskaidrojamās zināšanas, tās, kas apgūstamas skolā, lasot grāmatas vai klausoties profesoru lekcijas, vai konferences. Zināšanu apgalvojums ir tas, ko indivīdi skaidri formulē, izmantojot dabisku vai simbolisku valodu. Valoda ir būtiska sastāvdaļa emocionālās un garīgās pieredzes pārveidošanā par racionālām vai izteiktām zināšanām.

Eksistē dažādas zināšanu definīcijas. Vienu no visbiežāk citētajām zināšanu definīcijām formulējuši Tomass Davenport un Lorenss Prasaks (Davenport & Prusak, 1998): “Zināšanas ir pieredzes, vērtību, kontekstuālās informācijas un ekspertu ieskatu sajaukums, kas sniedz pamatu jaunas pieredzes un informācijas novērtēšanai un iekļaušanai. Organizācijās tās bieži tiek iestrādātas ne tikai dokumentos vai krātuvēs, bet arī organizācijas rutīnās, procesos, praksē un normās”.

Pētnieki zināšanas daļa arī pēc to formas, piemēram, Nonaka un Takeuchi definē zināšanas kā aisbergu - izteiktās jeb redzamās un neizteiktās jeb neredzamās zināšanās (Nonaka & Takeuchi, 1995b):

- izteiktas (angliski: explicit-knowledge) - racionālas zināšanas, kuras var formulēt, izmantojot jebkuru dabisku vai simbolisku valodu, viegli nododamas sociālā kontekstā;
- neizteiktas (angliski: tacit-knowledge) - personiskas un grūti formalizējamas, grūti nododamas citiem. Šajā zināšanu kategorijā ietilpst subjektīvie ieskati, intuīcijas un nojauta. Šīs zināšanas dziļi sakņojas indivīda darbībā un pieredzē, kā arī ideālos, vērtībās vai emocijās, kuras viņš vai viņa uzņem.

Zināšanas var uztvert arī kā objektus. Šādas interpretācijas formā Bolisani, Borgo un Oltra-mari (Bolisani et al., 2012) atzīmē: “Ja zināšanas var objektīvi noteikt, tad tās var

apstrādāt, reproducēt, uzglabāt un nodot, lielā mērā neatkarīgi no indivīda, kurš tās rada vai kam tās pieder”. Objektu zināšanas var iegult dokumentos, programmatūras kodos, datu bāzēs un dažādās platformās nododot tās koplietošanai starp darbiniekiem ar lielu varbūtību iegūt tādu pašu zināšanu interpretāciju.

Par zināšanu daļu var uzskatīt arī formāli definētu satura kopumu, ko var uzglabāt, indeksēt, izgūt un apstrādāt (Zack, 1999). Zināšanu elementi kā koncepti, raugoties no IKT (informāciju un komunikāciju tehnoloģijas) skatupunkta, tiek iedalīti sekojošās grupās (Zack, 1999; Osis, 2011):

- konceptuālie zināšanu elementi:
 - jēdzieni, kategorijas un definīcijas;
 - darbības, procesi un notikumu secības;
 - pamatojums darbībām vai secinājumi;
 - apstākļi un zināšanu lietošanas nodomi.
- zināšanu elementi no IKT skatupunkta:
 - daļas no dokumentiem, e-pasta ziņojumiem, audio datnes, video datnes, prezentācijas;
 - rekomendācijas, priekšlikumi, ekspertu viedokļi, definētas problēmas apraksti;
 - personīgas piezīmes;
 - ieguldījumi forumu, ziņu grupu vai tīmekļa emuāru papildināšanā;
 - uzglabāto pieredžu datu bāzu elementi;
 - dokumenti, saturoši produkta prezentācijas, labās un sliktās prakses, iegūtās mācības, pētījumus, stāstus, eksperimentu aprakstus, atskaites vai patentus;
 - prototipi;
 - modeļi, kuri satur vai attēlo procesus, datus, klases vai zināšanu struktūras;
 - apmācību objekti (definīcijas, skaidrojumi, formulas, uzdevumi vai eksāmena jautājumi);
 - prasmju apraksti;
 - zināšanu elementi, kas savieno iepriekšminētos zināšanu elementus ar indivīdiem, grupām, komandām vai organizācijas nodaļas (piemēram, konkrēta darbinieka vai organizācijas nodaļas prasmju apraksts);
 - iepriekšminēto elementu vērtējumi vai to komentāri.
- zināšanu elementi no lietotāju reakcijas datu skatupunkta:
 - sistēmas lietotāju apmeklētās vietnes;

- sistēmā veiktās darbības;
- sesiju ilgums;
- datu pieprasījumu regularitāte;
- sistēmas datu pieprasījumu metadati;
- sistēmas žurnālu datņu dati un kļūdu ieraksti.

Zināšanu definīcijas un tās jēdziena uztveres neviendabīgums rada plašas interpretāciju iespējas. Šī darba ietvaros zināšanas tiek interpretētas atbilstoši Ikujiro Nonaka un Hirotaka Takeuchi (Nonaka & Takeuchi, 1995b) organizācijas zināšanu radīšanas teorijai. Jaunu zināšanu izguve tiek piemērota tās lietderībai noteiktai organizācijai, saistot tās ar sociālo kontekstu, kurās zināšanas tiek radītas un lietotas.

1.1.1. ZINĀŠANU IZGUVE

Zināšanu izguve (angliski: knowledge discovery) ir tādu saistību atklāšanas process, kuru rezultātā no lielām datu kopām iegūstamas praktiskas zināšanas, izmantojot vienu vai vairākas tradicionālās datu izguves metodes, piemēram, tirgus grozu analīzi un kopu veidošanu (Loshin, 2013). Alternatīvi zināšanu izguve tiek definēta arī kā noderīgu zināšanu atklāšanas process no datu krājuma (Malheiro et al., 2018), kā arī jaunu izteiktu un/vai neizteiktu zināšanu izguve no datiem un informācijas vai no iepriekšēju zināšanu sintēzes (Schwartz & Te'eni, 2011).

Zināšanu izguves metodikas ir attīstījušās apvienojot statistikas pasauli un datorzinātnes. Zināšanu izguve galvenokārt koncentrējas uz zināšanu atklāšanu ar sešiem pamatuzdevumiem (Loshin, 2013):

- kopu veidošana un segmentēšana (angliski: clustering and segmentation), kuras uzdevums ir sadalīt lielu objektu kolekciju mazākās objektu grupās, kurām ir zināma līdzība;
- klasifikācija (angliski: classification), ietver konkrēta objekta atribūtu pārbaudi un piešķiršanu noteiktai klasei. Klasifikāciju var izmantot, lai sadalītu potenciālos klientus labākajos, viduvējos un mazvērtīgajos klientos, piemēram, lai atšķirtu aizdomīgus indivīdus lidostas drošības pārbaudē, identificētu krāpniecisku darījumu vai identificētu jauna pakalpojuma iespējas;
- novērtēšana (angliski: estimation), objekta kādas nepārtraukti novērotas skaitliskās vērtības piešķiršanas process. Piemēram, kredītriska novērtējums ne vienmēr ir binārs; tas var būt specifisks vērtējums, kas nosaka tieksmi neizpildīt saistības. Novērtējumu var izmantot kā daļu no klasifikācijas procesa (piemēram, izmantojot

novērtēšanas modeli, lai noteiktu indivīda gada ienākumus kā daļu no tirgus segmentēšanas procesa);

- prognozēšana (angliski: prediction), mēģinājums klasificēt objektus pēc kādas paredzamas uzvedības nākotnē. Klasifikāciju un novērtēšanu var izmantot prognozēšanai, izmantojot vēsturiskos datus, kur klasifikācija jau ir zināma modeļa veidošanai;
- interešu grupēšana (angliski: affinity grouping), saikņu vai asociāciju starp datu elementiem novērtēšanas process, kas nosaka kādu saikni starp objektiem;
- aprakstīšana (angliski: description), mēģinājums aprakstīt atklāto vai mēģināt izskaidrot zināšanu izguves procesa rezultātus.

Zināšanu izguve (minēta arī kā neuzraudzīta mācīšanās, angliski: unsupervised learning) izmanto nemarķētus, neklasificētus un kategorizētus apmācības datus (El Boucheffy & de Souza, 2020; Škoda & Adam, 2020). Neuzraudzītas mācīšanās galvenais uzdevums ir atklāt iepriekš neapzinātus un interesantus modeļus nemarķētos datos. Kopu veidošana ir izplatītākais neuzraudzītas mācīšanās algoritms, kuru pielieto, lai datus atrastu slēptus modeļus vai grupējumus. Kopu veidošanas piemēri ietver gēnu sekvences analīzi (angliski: gene sequence), tirgus izpēti un objektu atpazīšanu. Visbiežāk algoritmi, kurus pielieto neuzraudzītas mācīšanās procesiem, ietver kopu veidošanu, anomāliju noteikšanu (angliski: anomaly detection), neironu tīklus (angliski: neural networks) un latentu mainīgo modeļu apguvi (angliski: latent variable models) (El Boucheffy & de Souza, 2020; Škoda & Adam, 2020).

Šī darba ietvaros zināšanu izguve tiek interpretēta kā komplekss process, kura rezultātā no datu krājuma tiek iegūtas un/vai radītas lietderīgas izteiktas un/vai neizteiktas zināšanas.

1.1.2. VIEDĀS SISTĒMAS

Vārda "vieds" plašais pielietojums, lai apzīmētu vai kvalificētu noteiktus sistēmu veidus, nenodrošina universālu to funkciju identifikāciju, kuras padarītu konkrētu sistēmu viedu (Romero et al., 2020). Atsevišķi pētnieki uzskata, ka viedums ir iespējams plaši pielietojot IKT un digitālās iespējas (Garau, 2014; Kanter & Litow, 2009), pastāv arī visaptverošāki viedokļi par vieduma jēdzienu, uzskatot, ka viedums tiek sasniegts, izmantojot vairāku faktoru, piemēram, politikas, ekonomikas, pārvaldības, izglītības, indivīdu, tehnoloģiju, ilgtspējas u.c., sadarbību, lai uzlabotu noteiktu vidi un/vai pilnveidotu iedzīvotājus (Cavada et al., 2014; Douaioui et al., 2018; Nam & Pardo, 2011).

Biznesa un vadības sfērā viedumu nosaka biznesa stratēģija, ieinteresēto personu pieredze un/vai uzvedība, kā arī tirgus apguves stratēģija (Romero et al., 2020). Sociālo zinātņu un izglītības vidē viedums tiek uzskatīts par sociālu konstrukciju nevis indivīdam raksturīgām spējām (Dunleavy, 2018).

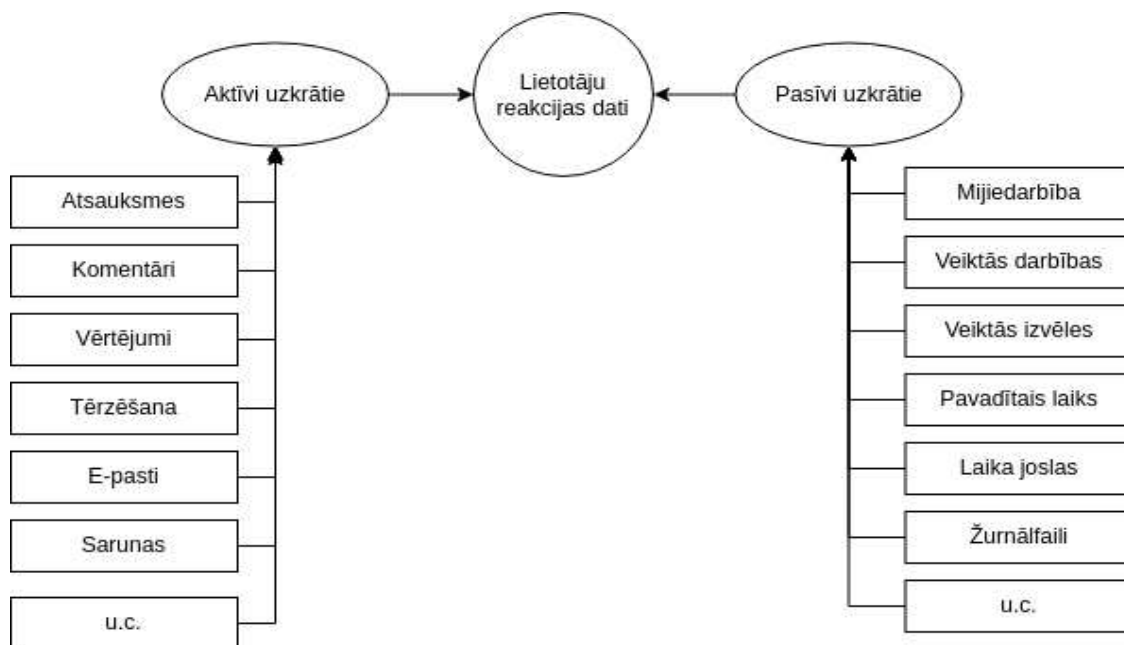
Plašāk lietotā sistēmas definīcija: “sistēma ir savstarpēji savienotu daļu kopums ar īpašībām, kas ir lielākas par šo daļu īpašību summu. Sistēmai ir robežas un atribūtu kopums, kas padara to unikālu salīdzinājumā ar citām sistēmām (Naudet et al., 2009)”. Sistēmu raksturojošie pamatelementi tiek izšķirti sekojoši (Naudet et al., 2009):

- sistēmas elements: sistēmas komponente;
- saistība (relācija): saikne starp divām sistēmas vienībām;
- mērķis: darbību kopums, ko sistēma spēj veikt noteiktā vidē, lai sasniegtu izvirzīto mērķi;
- vide: viss, kas atrodas ārpus sistēmas robežām;
- saskarne: sistēmas elements, caur kuru tiek izveidots savienojums starp sistēmu un tās vidi.

Identificētie sistēmas pamatelementi tiek uzskatīti par nepieciešamiem jebkurai sistēmai. Viedās sistēmas ietver visus šos pamatelementus un pievieno vēl viedajām sistēmām raksturīgākos elementus un spējas. Raksturīgākie viedas sistēmas elementi un spējas, balstoties uz Romero (Romero et al., 2020):

- komunikācijas spēja: sistēmas elementi spēj mijiedarboties, lai apmainītos ar datiem, paziņotu elementu darbības un informētu par apkārtējo stāvokli;
- iegultas zināšanas: sistēma spēj aptvert cilvēku / ekspertu pieredzi un kompetenci, kā arī informāciju par vidi;
- mācīšanās spēja: mācīšanās ietver dažādu metožu un algoritmu izmantošanu, kas ļauj modificēt sistēmas zināšanas;
- spēja spriest: viedās sistēmas spēj pamatot savu zināšanu izmantošanu lēmumu pieņemšanā;
- uztveres spēja: tādu elementu klātbūtne, ar kuru mijiedarbojas viedā sistēma. Lai varētu veikt šo mijiedarbību, tiek izmantoti sensori, kas uztver apkārtējās vides stāvokli, kā arī izpildmehānismi, kuri spēj veikt darbības, kuras pielāgo šo stāvokli;
- kontroles spēja: viedās sistēmas tiek veidotas, lai palīdzētu to lietotājiem vai nodrošinātu nepieciešamos rīkus un mehānismus, lai uzlabotu lietotāju dzīves aspektus;

darbiniekiem (skat., 1.2.att.). Kā arī pasīvi uzkrātie lietotāju reakcijas dati, mijiedarbības dati ar sistēmu, piemēram, veiktās darbības sistēmā, veiktās izvēles, pavadītais laiks u.c. (Kechagia, Mitropoulos & Spinellis, 2015).



1.2.attēls. Lietotāju reakcijas dati (autora veidos).

Darba ietvaros par lietderīgāku tiek uzskatīti pasīvi uzkrātie dati. Šī veida datus iespējams ievākt par sistēmas lietotājiem, iekļaujot tos lietotājus, kuri neņem dalību lietotāja ievadu (aktīvi uzkrātie lietotāju reakcijas dati) pieprasošās darbībās. Pasīvi tiek uzkrāta arī sistēmu raksturojošā informācija - kļūdas, diagnostikas, sistēmas atbildes laiki, atsevišķu komponentu pieprasījumi u.c. Pasīvi uzkrātie sistēmas dati var nodrošināt efektīvāku un datus balstītu lēmumu pieņemšanas procesu, piemēram, hipotēžu validācijai, lai noteiktu kāda izskata komponentes strādā labāk vai kādi kvalitātes kritēriji ir būtiski sistēmu lietotājiem (Dugar, 2019).

Pasīvo datu priekšrocība ir tā, ka tie nav balstīti uz sistēmas lietotāju veiktām aktivitātēm, kas var būt pakļautas gan nepilnīgai dalībai, gan mērķtiecīgai darbībai. Vēl kāda pasīvās datu vākšanas priekšrocība ir tā, ka tā neapšaubāmi samazina respondentu slogu, novēršot vajadzību atbildēt uz jautājumiem par savu digitālo uzvedību noteiktā sistēmā (Barthel et al., 2020).

Pasīvi uzkrāto datu (telemetrijas) ieguves process iekļauj tādas darbības kā atbilstošo metriku un tām nepieciešamo datu kopu noteikšanu, lietderīgāko instrumentu noteikšanu (angliski: instrumentation), un datu pārraides, un uzglabāšanas risinājumu

identificēšanu. Sistēmu telemetrijas jeb pasīvi uzkrāto datu process iekļauj arī tādas darbības kā datu apstrāde un atspoguļošana (Dugar, 2019).

Sistēmu uzkrātajiem pasīvajiem lietotāju reakcijas datiem ir nepieciešams arī noteikt balansu datu vienības pārsūtīšanas ātrumam un pārraides izmaksām (tīkla, CPU, GPU lietojuma, glabātuves). Neatkarīgi no tehniskajiem ierobežojumiem, kas saistīti ar jebkuru programmatūru, pasīvi uzkrāto datu ievākšanas procesā ir jāveic vairākas profilēšanas - tīkla (angliski: network), iekārtu (angliski: device), programmatūras (angliski: mobile/standalone), pārraides biežumu (angliski: transmission frequency). Profilēšana sniedz iespēju izvirzīt korektāku pasīvās telemetrijas datu uzkrāšanas biežumu un uzkrājamo parametru prasības noteiktai sistēmai (Dugar, 2019).

Jāņem vērā, ka gan pasīvajos datos, gan aktīvi uzkrātajos datos būs identificējami trūkumi. Tos uzrāda Barthel et al. (Barthel et al., 2020) veiktais pētījums, kur pasīvi ģenerēti aprēķini par to indivīdu īpatsvaru, kuri apmeklējuši ziņu vietnes, ir zemāki, salīdzinājumā ar aktīvi ievāktajiem datiem. Savukārt vairākas uz aptauju (aktīvi ievāktajiem datiem) balstītas aplēses, iespējams, ir palielinātas ar mērķtiecīgu indivīdu reaģēšanu (sociālās vēlamības novirze / angliski: social desirability bias).

Pētniecības viedokļu apkopošana par tādiem traktātiem kā zināšanas, viedās sistēmas un lietotāju reakcijas dati sniedz informāciju par promocijas darbā iekļautajām sfērām un to interpretāciju. Savukārt, lai noteiktu aktuālās un pieejamās teorijas un metodoloģijas zināšanu izguves ietvaru arhitektūras modelēšanā un tehnoloģiju aprobācijā, darba autors ir izstrādājis saistošo pētījumu atlases metodoloģiju, kura uzrādīta 1.2 nodaļā.

1.2. SAISTOŠO PĒTĪJUMU ATLASE

Promocijas darba teorētiskās daļas literatūras atlasei darba autors vadās pēc zemāk uzrādītās meklēšanas stratēģijas. Meklēšanas stratēģija ir sakārtota galveno terminu struktūra, kas tiek izmantota, lai meklētu izvēlētajās datu bāzēs. Meklēšanas stratēģija apvieno meklēšanas jautājuma galvenos jēdzienus, kas sniedz precīzu rezultātu iegūšanu. Meklēšanas stratēģijas izveide iekļauj (Elsevier, 2022; University of Leeds, 2022; University of Wollongong Australia, 2022):

- iespējamo meklēšanas terminu definēšanu;
- atslēgvārdu un frāžu noteikšanu;
- meklēšanas vienumu saīsinātus un koriģētus variantus;
- apskatāmo tēmu virsrakstu noteikšanu.

Pamatojoties uz augstāk minēto stratēģiju, veiksmīgai saistošas un atbilstošas informācijas atrašanai noteiktā sfērā, lietderīgi izmantot strukturētu pieeju. Aplūkojamās tēmas ietvaros tiek izšķirti galvenie jēdzieni un katram jēdzienam definēti atslēgvārdi. Atslēgvārdu saraksta paplašināšanai tiek pielietota izvērsta meklēšana, izmantojot sinonīmus un alternatīvas frāzes. Meklēšanā iekļauti termini, lai identificētu un norādītu to nozīmīgumu attiecīgajā jomā. Pielietojot šo pieeju iespējams noteikt vai termins ir attiecināms pret darbā aplūkojamo tēmu, vai arī nepieciešams rast precīzākus terminus.

Darba ietvaros pētījumu meklēšanas stratēģija dalīta vairākās iterācijās sākotnēji definējot uzdevumus (pilna apjoma meklēšanas stratēģiju skat., I pielikumā). Meklēšanas stratēģijai izvirzītie uzdevumi:

- noteikt prominentas publikācijas:
 - publikāciju autorus;
 - domēnus un problēmjomas, kuras apskata publikācijās;
 - kādus instrumentus izmanto zināšanu izguve;
 - kāda veida zināšanas tiek iegūtas.
- izveidot pārskatu par iegūto informāciju (skat., I pielikums);
- veikt literatūras atlasi zinātniskajās datubāzēs;
- apkopot informāciju par publikācijām:
 - autora/u pētījumu virziens;
 - saistītās prominentu autoru publikācijas.

Darba ietvaros meklēšanas stratēģijai izvirzītie termini:

- zināšanu izguve (angliski: knowledge discovery);
- lietotāju reakcijas dati (angliski: response feedback data);
- viedās sistēmas (angliski: smart systems).

Lai meklēšanas stratēģijas rezultātus būtu iespējams apkopot un tos ierobežotu apjomā, izvirzīti noteikti kritēriji - pieci (5) meklēšanas scenāriji (augstākie desmit rezultāti katrā scenārijā):

1. visvairāk citētās publikācijas ar noteikto atslēgvārdu, kopš atslēgvārds pirmoreiz parādās zinātniskajā datubāzē;
2. visvairāk citētās publikācijas pēdējos desmit gados, lai noteiktu darbus, kurus pētnieki pielieto;
3. pētnieki ar visvairāk publikācijām un viņu virziens (kopš 1991.gada), laika ierobežojums 30 gadi;
4. aktīvākie pētnieki pēdējos desmit gados un viņu pētījumu virziens;

5. saistošās publikācijas (autora subjektīvais ierobežojums) no augstākajiem 100 rezultātiem pēdējos piecos gados.

Meklēšanas stratēģijas piecu scenāriju rezultātus pilnā apjomā iespējams aplūkot pielikumā (skat., I pielikums), ierobežota apjoma meklēšanas scenāriju rezultāti tiek uzrādīti 1.1.tabulā.

1.1. tabula. Meklēšanas scenāriju rezultāti (fragments, rezultātus skat., I pielikums) (autora veidota)

Atslēgvārds	Meklēšanas scenārijs	Rezultāti / Publikācijas (paraugs)
Zināšanu izguve (angliski: knowledge discovery)	1. Visvairāk citētās publikācijas	Sarwar, B., Karypis, G., Konstan, J., Riedl, J. Item-based collaborative filtering recommendation algorithms 2001 (5262 cited)
	2. Visvairāk citētās publikācijas pēdējos 10 gados	Han, J., Kamber, M., Pei, J., Data Mining: Concepts and Techniques (Book), 2012 (3365)
	3. Autori ar visvairāk publikācijām un viņu virziens (kopš 1991.gada),	Holzinger A.: Biomedicine, Bioinformatics, 50 documents
	4. Aktīvākie autori pēdējos 10 gados	Holzinger A.: Biomedicine, Bioinformatics, 50 documents
	5. Saistošās publikācijas	Ge, Z., Song, Z., Ding, S.X., Huang, B. Data Mining and Analytics in the Process Industry: The Role of Machine Learning, 2017
Response feedback data (lietotāju reakcijas dati)	1. Visvairāk citētās publikācijas	Huang, Z., Chen, H., Zeng, D., Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering, 2004 (473)
	2. Visvairāk citētās publikācijas pēdējos 10 gados	Khan, S., Yairi, T., A review on the application of deep learning in system health management, 2018 (306)
	3. Autori ar visvairāk publikācijām un viņu virziens (kopš 1991.gada),	Sawahashi, M., W-CDMA, last in 2005, 6 documents
	4. Aktīvākie autori pēdējos 10 gados	Karatzoglou, A., Context aware representations, preference learning, mobile app discovery, 5 documents, latest from 2014
	5. Saistošās publikācijas	Bayer, I., He, X., Kanagal, B., Rendle, S., A generic coordinate descent framework for learning from implicit feedback, 2017

Atslēgvārds	Meklēšanas scenārijs	Rezultāti / Publikācijas (paraugs)
Smart systems (viedās sistēmas)	1. Visvairāk citētās publikācijas	Hossain, M.S., Muhammad, G., Cloud-assisted Industrial Internet of Things (IIoT) - Enabled framework for health monitoring, 2016, (353)
	2. Visvairāk citētās publikācijas pēdējos 10 gados	Hossain, M.S., Muhammad, G., Cloud-assisted Industrial Internet of Things (IIoT) - Enabled framework for health monitoring, 2016, (353)
	3. Autori ar visvairāk publikācijām un viņu virziens (kopš 1991.gada),	Fummi, F., smart system design, 14 documents, active
	4. Aktīvākie autori pēdējos 10 gados	Fummi, F., smart system design, 14 documents, active
	5. Saistošās publikācijas	Hatcher, W.G., Yu, W., A Survey of Deep Learning: Platforms, Applications and Emerging Research Trends, 2018

Meklēšanas scenārijos noteikta kopa ar potenciāli saistošām publikācijām un pētniekiem (skat., 1.1.tabula un I pielikums). Meklēšanas scenāriju rezultāti apstrādāti manuāli, jo kvantitatīva rezultātu apkopošana nesniedza pietiekamu informāciju par pētījumiem un to vai šie pētījumi sniedz pievienoto vērtību darba ietvaros.

Saistošo pētījumu novērtēšanai darba ietvaros izvirzīti sekojoši nosacījumi, realizēti iepazīstoties ar publikāciju saturu (skat., I pielikums):

- pētījuma sfēra;
- pētījuma uzdevums;
- kādu pētnieku darbi tiek izmantoti;
- kādi ir ieteiktie nākotnes uzdevumi;
- aprakstītās nākotnes tendences.

Pētījumu atbilstības novērtēšanai darba autors izvirza vienkāršotu kategorizācijas skalu, kura, balstoties uz augstāk minētajiem nosacījumiem, pielietota, lai meklēšanas scenāriju rezultātus šķirotu atbilstoši to potenciālajai pievienotajai vērtībai (skat., I pielikums):

- 1 - pētījumā tiek aplūkota cita sfēra, promocijas darbam nebūs ieguvumu;
- 2 - daļēji atbilstoša sfēra un uzdevums;
- 3 - atbilstoša sfēra un uzdevumi, nepieciešama padziļināta izpēte.

Darbi, kuri atbilstoši kategorizācijas skalai, novērtēti ar trīs, apkopoti rezultātu tabulā iekļaujot saistītos darbus, kuri potenciāli var sniegt pievienoto vērtību promocijas darba izstrādē (skat., 1.2.tabulu un I pielikumu).

Piezīme: Literatūras izpētes procesā apkopie pētījumi promocijas darba literatūras sarakstā netiek iekļauti, ja promocijas darbā tiem nav tiešas atsauces.

1.2. tabula. Literatūras izpētes, rezultātu apkopojums (fragments, rezultātus skat., I pielikums) (autora veidota)

Publikācija	Saistītie, potenciāli lietderīgie pētījumi
Osman, A.M.S. A novel big data analytics framework for smart cities, 2019	<ul style="list-style-type: none"> ● Tsai, C.-W., Lai, C.-F., Chao, H.-C., Vasilakos, A.V., Big data analytics: a survey, 2015 ● Hashem, I.A.T., Chang, V., Anuar, N.B., Adewole, K., Yaqoob, I., Gani, A., Ahmed, E., (...), Chiroma, H., The role of big data in smart city, 2016 ● Al Nuaimi, E., Al Neyadi, H., Mohamed, N., Al-Jaroodi, J., Applications of big data to smart cities, 2015 ● Chen, M., Mao, S., Liu, Y., Big data: A survey, 2014 ● Singh, D., Reddy, C.K., A survey on platforms for big data analytics, 2015 ● Santana, E.F.Z., Chaves, A.P., Gerosa, M.A., Kon, F., Milojevic, D.S., Software platforms for smart cities: Concepts, requirements, challenges, and a unified reference architecture, 2017 ● Cooper, H.M., Organizing knowledge syntheses: A taxonomy of literature reviews, 1988 ● Strohbach, M., Ziekow, H., Gazis, V., Akiva, N., Towards a Big Data Analytics Framework for IoT and Smart City Applications, 2015 ● Khan, Z., Anjum, A., Soomro, K., Tahir, M.A., Towards cloud based big data analytics for smart future cities, 2015
Chen, M., Herrera, F., Hwang, K., Cognitive Computing: Architecture, Technologies and Intelligent Applications, 2018	<ul style="list-style-type: none"> ● Jin, J., Gubbi, J., Marusic, S., Palaniswami, M., An information framework for creating a smart city through internet of things, 2014 ● Catarinucci, L., De Donno, D., Mainetti, L., Palano, L., Patrono, L., Stefanizzi, M.L., Tarricone, L., An IoT-Aware Architecture for Smart Healthcare Systems, 2015 ● Chen, M., Mao, S., Liu, Y., Big data: A survey, 2014

Publikācija	Saistītie, potenciāli lietderīgie pētījumi
	<ul style="list-style-type: none"> • Chen, M., Tian, Y., Fortino, G., Zhang, J., Humar, I., Cognitive Internet of Vehicles, 2018 • Fernández, A., del Río, S., López, V., Bawakid, A., del Jesus, M.J., Benítez, J.M., Herrera, F., Big Data with Cloud Computing: An insight on the computing environment, MapReduce, and programming frameworks, • Chen, M., Qian, Y., Hao, Y., Li, Y., Song, J., Data-Driven Computing and Caching in 5G Networks: Architecture and Delay Analysis, 2018 • Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., (...), Hassabis, D., Mastering the game of Go with deep neural networks and tree search, 2016

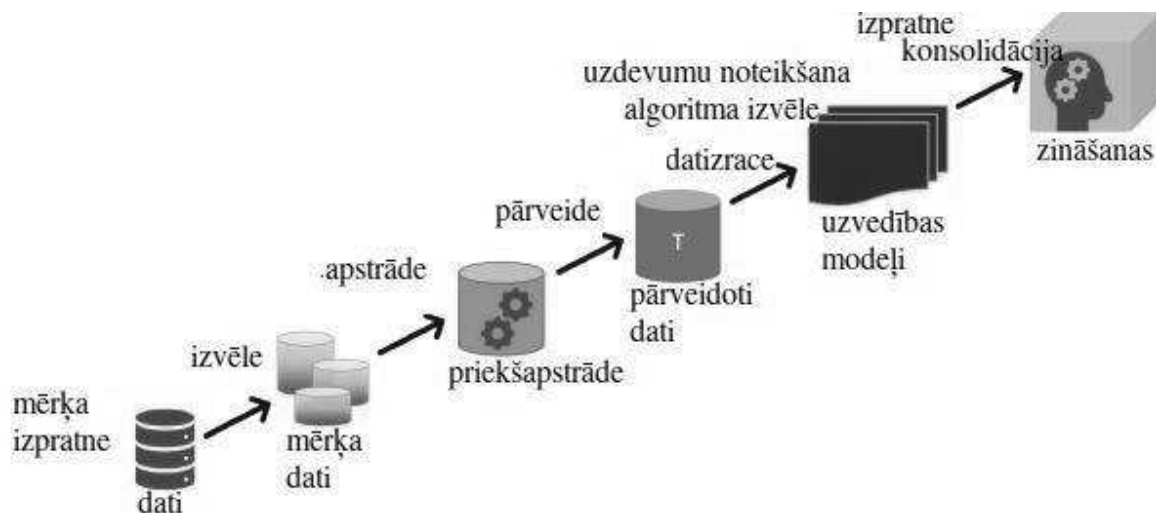
Strukturētā pieeja literatūras izpētei un apkopošanai ieviesta promocijas darba izstrādes procesā palielinot varbūtību pielietot pētījumus, kuri ir nozīmīgi un aktuāli. Ņemts vērā, ka arī izmantojot subjektīvu literatūras izpēti, pastāv iespēja neievērot fundamentālus pētījumus un to autorus. Lai samazinātu šo varbūtību, darbā tiek izmantoti gan uzrādītie literatūras izpētes rezultāti (skat., I pielikums), gan arī darba izstrādes procesā identificētie pētījumi. Saistošo pētījumu meklēšanas stratēģijas pielietošanas un rezultātu apkopošanas procesā izstrādātais promocijas darbam pielietojamās literatūras apkopojums izmantots, lai identificētu aprakstītos zināšanu izguves procesu modeļus un zināšanu izguves ietvarus.

1.3. ZINĀŠANU IZGUVES PROCESS

Termins zināšanu izguve (angliski: knowledge discovery) attiecas uz zināšanu izguves procesu datus un akcentē konkrētu datu izguves metožu augsta līmeņa pielietojumu. Process ir saistošs pētniekiem mašīnmācīšanās, modeļu atpazīšanas, datu bāzu, statistikas, mākslīgā intelekta, ekspertu sistēmu un datu vizualizācijas jomās. Zināšanu izguves procesa mērķis ir iegūt zināšanas no datiem. Zināšanas tiek iegūtas, izmantojot datu izguves algoritmus, lai saskaņā ar definētiem nosacījumiem un specifikācijām iegūtu zināšanas, izmantojot datubāzes, kā arī piemērotu nepieciešamās datu priekšapstrādes, apakšizlases un transformācijas (Technopedia Inc, 2017).

Zināšanu izguves jomā 90. gados pētnieki definēja daudzpakāpju procesu, kurā instruēja datu ieguves rīku izmantotājus procesu izpildes centienos (Škoda & Adam, 2020).

Definētā procesa pamatzudevums ir noteikt darbību secību, kas palīdzētu zināšanu izguves procesam brīvi izvēlētā domēnā. Viens no pirmajiem piedāvātajiem zināšanu izguves modeļiem publicēts 1996.gadā, modelis zināms kā zināšanu izguves process (angliski: KDD process) (Fayyad et al., 1996). Deviņu soļu modelis, ko izstrādājuši Fayyad et al., tiek uzskatīts par vienu no galvenajiem modeļiem zināšanu izguves procesā (Osei-Bryson & Barclay, 2015).



1.3.attēls. Zināšanu izguves procesa modelis (adaptēts no Fayyad et al., 1996).

KDD process (skat., 1.3.att.) iekļauj sekojošos soļus (Fayyad et al., 1996):

1. pētāmās jomas definēšana un izpratne, atbilstošu priekšzināšanu apgūšana, lietotāja mērķu identificēšana (ievade: risināmā problēma/mērķis, izvade: problēmas/domēna/mērķa izpratne);
2. mērķa datu kopas izveide – datu kopas atlase, apakškopas mainīgo datu atribūtu identificēšana un datu paraugu izveide (izeja: mērķa dati/datuma kopa);
3. datu attīrīšana un priekšapstrāde – noviržu un neatbilstošu datu noņemšana, iztrūkstošo datu apstrāde, zināmo datu izmaiņu identificēšana (izeja: apstrādāti dati);
4. datu pārveide – noderīgu funkciju atrašana, kas attēlo datus atbilstoši izvirzītajam mērķim (izeja: pārveidoti dati);
5. uzdevuma noteikšana – lēmums par to, kādas metodes piemērot klasifikācijai, klasterizācijai, regresijai vai citam uzdevumam (izeja: izvēlētās metodes);

6. algoritmu izvēle – metodes noteikšana modeļu meklēšanai, pieņemot lēmumu par piemērotiem modeļiem un to parametriem, kā arī saskaņojot metodes ar procesa mērķi (izeja: atlasītie algoritmi);
7. datizrace — atbilstošo modeļu meklēšana noteiktā formā, piemēram, klasifikācijas noteikumi, lēmumu koki, regresijas modeļi, tendences, kopas un asociācijas (izeja: modeļi);
8. iegūto datu interpretācija – modeļu izpratne un vizualizācija, pamatojoties uz iegūtajiem modeļiem (izeja: interpretētie modeļi);
9. zināšanu konsolidācija – atklāto modeļu izmantošana KDD procesa analizētā sistēmā, zināšanu dokumentēšana un ziņošana lietotājiem, kā arī nepieciešamības gadījumā konfliktu pārbaude un risināšana (izeja: zināšanas, darbības/lēmumi, pamatojoties uz rezultātiem).

KDD process ir vienkārša metodoloģija un salīdzinoši vienkārši izprotams modelis zināšanu izguvei. Pētnieki uzskata, ka šim modelim ir divi būtiski trūkumi (Škoda & Adam, 2020). Pirmkārt, pirmie definētie līmeņi ir pārāk abstrakti un nav skaidri un formalizēti. Detalizācijas trūkums mainīts vēlākajās modeļa versijās, izmantojot formalizētākus soļu aprakstus (dažos gadījumos izmantojot standartus, procesu automatizāciju vai rīkus un platformas). Otrs trūkums ir biznesa aspektu apraksta trūkums, jo modeļa izstrāde sākumā bija pētniecībā balstīta pieeja.

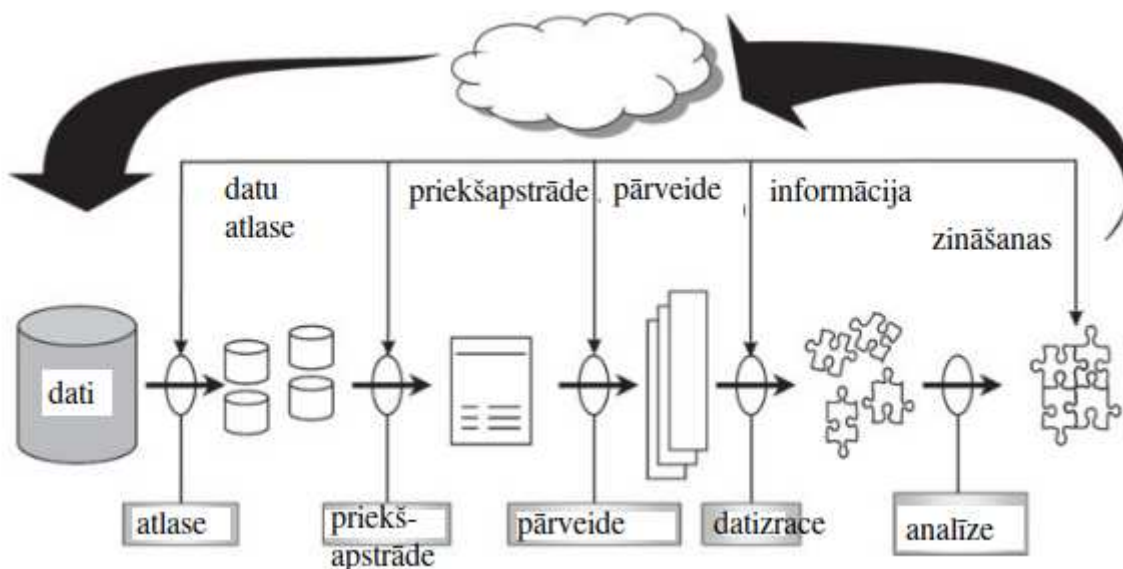
Zināšanu izguves procesu var vienkāršot vairākās pamata aktivitātēs, kuras iekļautas vispārējos projektu vadības pamatprincipos zināšanu izguves projektos (Osei-Bryson & Barclay, 2015):

1. skaidras izpratnes veidošana par organizācijas mērķiem un uzdevumiem konkrētam projektam - ko organizācija vēlas atklāt no datiem, ko vēlas sasniegt ar projektu;
2. saistošo datu identificēšana, analīze un sagatavošana, lai atvieglotu datorizētu ieguvu;
3. algoritmu pielietošana, lai no datiem iegūtu saistošas zināšanas;
4. piemērotāko modeļu izvērtēšana un atlase, pamatojoties uz sākotnēji izvirzītajiem mērķiem;
5. iegūto zināšanu pielietošana un izmantošana.

Minētie soļi ir būtiski un nepieciešami priekšnoteikumi veiksmīga projekta īstenošanai. Rezultātā ir izveidoti vairāki procesu modeļi, kas izstrādāti gan akadēmiskajā, gan nozares vidē.

1.3.1. PIECU SOĻU MODELIS

Piecu soļu zināšanu izguves modeli pirmoreiz piedāvāja Cabena et al. 1998. gadā (Cabena, 1998). Piecu soļu modelis iekļauj biznesa mērķu noteikšanu, datu sagatavošanu, datu izgūvi, rezultātu analīzi un iegūto zināšanu pielietojšanu (skat., 1.4.att.).



1.4. attēls. Zināšanu izguves cikls, piecu soļu modelis (adaptēts no Cabena, 1998).

Detalizēts piecu soļu modeļa darbības apraksts (Cabena, 1998):

1. biznesa mērķu noteikšana. Solis ietver noteiktu uzņēmējdarbības problēmu vai izaicinājumu definēšanu. Minimālās prasības soļa izpildei ir definēta biznesa problēma vai potenciāla iespēja. Sagaidāmais rezultāts visbiežāk arī tiek noteikts šajā procesa solī;
2. datu sagatavošana. Pētnieki (Cabena, 1998) atzīmē, ka datu sagatavošana ir resursietilpīgākais procesa solis, kas parasti prasa līdz 60% no visa projekta laika. Šis solis ietver arī apakšuzdevumus:
 - a. datu atlasī: ietver visu iekšējo vai ārējo informācijas avotu identificēšanu un atbilstošās datu apakškopas atlasī;
 - b. datu priekšapstrādi: datu priekšapstrāde ietver datu kvalitātes izpēti, lai sagatavotu tos turpmākai analīzei un noteiktu darbību veidus, kas būs potenciāli iespējami;
 - c. datu transformāciju: datu transformācijas laikā iepriekš apstrādātie dati tiek pārveidoti, lai izveidotu analītisko datu modeli. Šis analītiskais modelis ir informatīvs un atspoguļo atlasīto datu

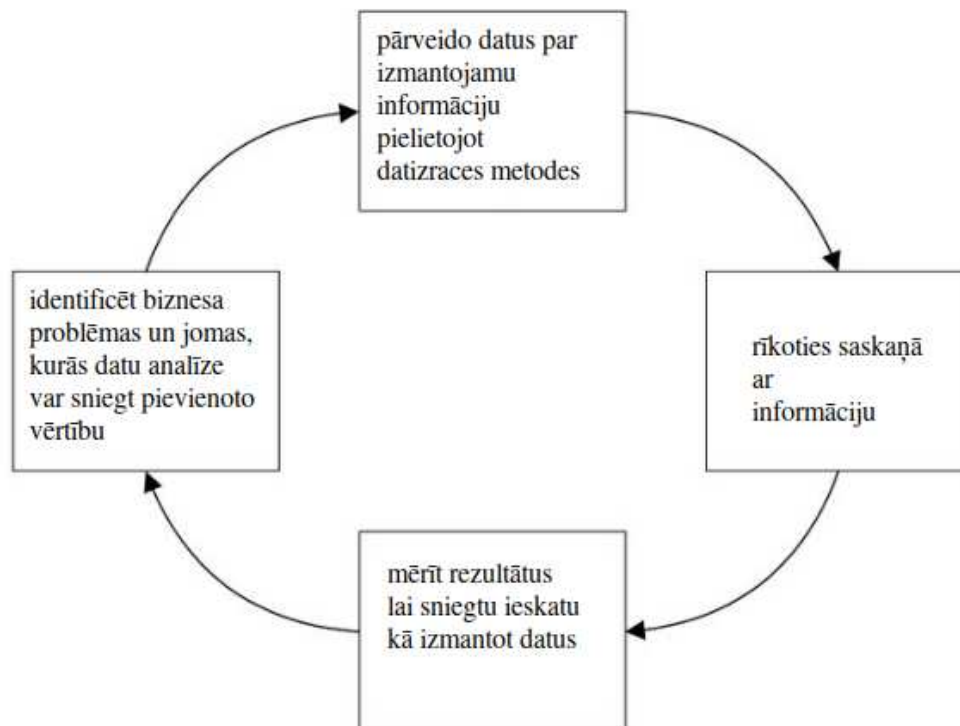
konsolidētu, integrētu un laika atkarīgu pārstrukturēšanu. Datu transformācija ir būtisks uzdevums, jo zināšanu izguves procesa rezultāts ir atkarīgs no tā, kā datu analītiķis/i izlemj strukturēt un izveidot datu ievadi.

3. datu izguve. Soļa mērķis ir skaidri piemērot atlasīto algoritmu vai algoritmus iepriekš apstrādātajiem datiem. Faktiskie uzdevumi atšķiras atkarībā no rezultāta veida, kas definēts biznesa mērķa noteikšanas solī. Paredzamā modeļa izstrāde ieplānota kā ciklisks process, kurā modeļi tiek atkārtoti apmācīti, pamatojoties uz izlases datiem, pirms tie tiek pārbaudīti;
4. rezultātu analīze: Piecu soļu procesa modelī rezultātu analīze nav atdalāma no datu izguves posma. Konkrētie uzdevumi šajā solī ir atkarīgi no uzdevuma veida, kas definēts biznesa mērķa noteikšanas solī. Soļa mērķis paliek nemainīgs - interpretēt un novērtēt datu izguves posma rezultātu;
5. zināšanu pielietošana: Šis solis noslēdz zināšanu izguves procesu, kas tika uzsākts, kad biznesa mērķi tika definēti procesa sākumā. Šī soļa mērķis ir īstenot sākuma posmā definētos uzdevumus saskaņā ar jaunajām un pielietojamajām zināšanām, kuras iegūtas iepriekšējos procesa soļos. Šajā solī divi galvenie izaicinājumi ir sekojoši: pārliecinoši, uz uzņēmējdarbību vērstā veidā, prezentēt jaunās zināšanas un formulēt veidus, kādos jaunās zināšanas var pielietot.

1.3.2. ČETRU SOĻU MODELIS

Četru soļu modeli piedāvā pētnieki Berijs un Linofs (Berry & Linoff, 1997), modelis sastāv no sekojošiem soļiem (skat., 1.5.att.):

1. problēmas identificēšana;
2. problēmas analīze;
3. nepieciešamo darbību izpilde;
4. rezultātu mērīšana un novērtēšana.



1.5.attēls. Zināšanu izguves cikls, četrus soļus modelis (adaptēts no Berry & Linoff, 1997).

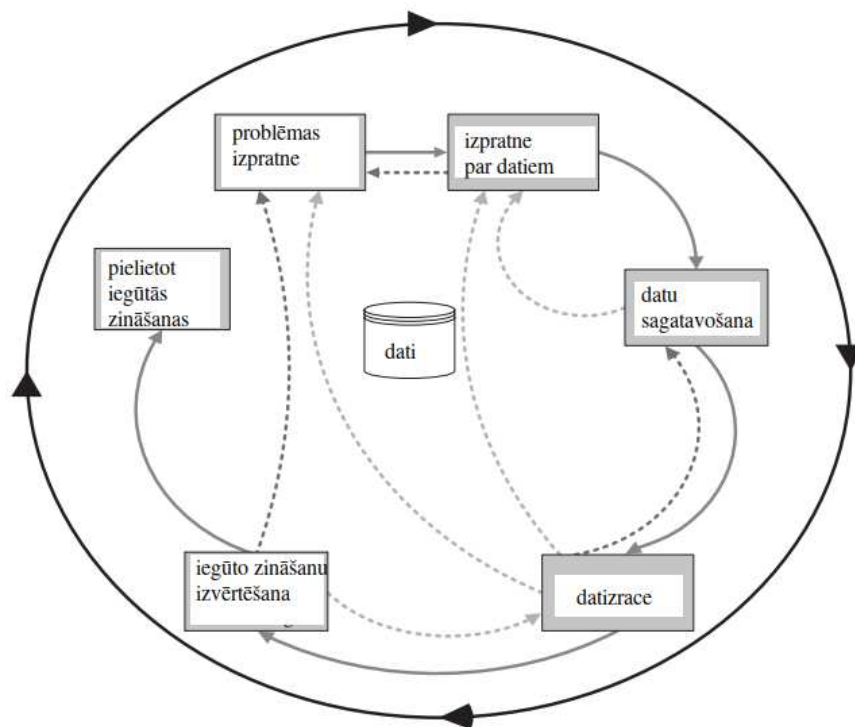
Berry un Linoff (Berry & Linoff, 1997) precizēja 11 posmus, sīkāk detalizējot piedāvāto četrus soļus pieeju:

1. pārveidot biznesa problēmu par datu ieguves problēmu;
2. izvēlēties atbilstošos datus;
3. iepazīties ar datiem;
4. izveidot modeļu komplektu;
5. novērst problēmas ar datiem;
6. pārveidot datus, lai informācija kļūtu izprotama;
7. veidot modeļus;
8. novērtēt modeļus;
9. izvietot modeļus;
10. novērtēt rezultātus;
11. atkārtot procesu no sākuma.

Četrus soļus modelis paredzēts kā iteratīvs process, ar mērķi zināšanām piešķirt praktisku pielietojumu noteiktas organizācijas vajadzībām.

1.3.3. SEŠU SOĻU MODELIS

2005.gadā pētnieki Cios un Kurgan piedāvāja sešu soļu zināšanas izguves modeli (skat. 1.6.att.). Tas iekļauj problēmas sfēras izpratni, datu izpratni, datu sagatavošanu, datu izguvi, rezultātu analīzi un iegūto zināšanu pielietojumu (Cios & Kurgan, 2005).



1.6. attēls. Zināšanu izguves cikls, sešu soļu modelis (adaptēts no Cios & Kurgan, 2005).

Sešu soļu modelim sniegti arī detalizētāki procesu apraksti (Cios & Kurgan, 2005):

1. problēmas izpratne. Šajā solī nepieciešams sadarboties ar nozares ekspertiem, lai definētu problēmu un noteiktu projekta mērķus, identificētu galvenos iesaistītos indivīdus, un uzzinātu par aktuālajiem problēmas risinājumiem. Ietver domēna specifiskās terminoloģijas apgūšanu. Rezultātā tiek izveidots problēmas apraksts, uzskaitot arī ierobežojumus;
2. izpratne par datiem. Solis ietver datu paraugu iegūvi un atbilstošu datu atlasīšanu. Tiek pārbaudīta datu lietderība saistībā ar zināšanu izguves mērķiem. Tiek pārbaudīts datu pilnīgums, dublēšana, trūkstošās vērtības un atribūtu vērtību ticamība;
3. datu sagatavošana. Datu sagatavošanas solis nosaka to, cik veiksmīgs būs zināšanu izguves process. Parasti solis aizņem apmēram pusi no visa projekta laukietilpības. Šajā solī tiek pieņemti lēmumi par to, kuri dati tiks

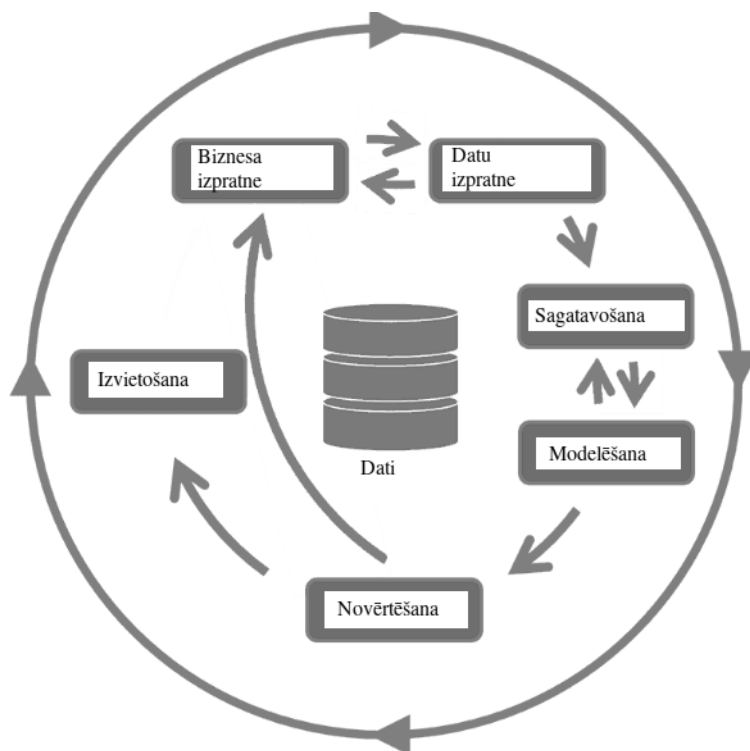
izmantoti kā ievads. Var ietvert datu paraugu iegūvi, korelācijas un nozīmīguma testu veikšanu, datu tīrīšanu, piemēram, datu ierakstu pilnīguma pārbaudi un trokšņu noņemšanu vai labošanu. Koriģētos datus var tālāk apstrādāt, izmantojot funkciju atlasē un ieguves algoritmus, iegūstot jaunus atribūtus un grupējot datus;

4. datizrace. Tiek uzskatīts par būtiskāko posmu tieši zināšanu izguves procesā. Šis solis ietver plānoto rīku izmantošanu, daudzu veidu algoritmus, piemēram, aptuvenās un izplūdušās kopas, Bajeza metodes (Bernardo, 2011), evolucionāro skaitļošanu, mašīnmācīšanos, neironu tīklus, klasterizāciju, un priekšapstrādes metodes. Šis solis ietver vairāku datu izguves rīku izmantošanu datiem, kas sagatavoti trešā procesa soļa laikā. Pirmkārt, tiek izstrādātas apmācības un testēšanas procedūras un izveidots datu modelis;
5. iegūto zināšanu izvērtēšana. Solis ietver rezultātu izpratni un pārbaudi - vai iegūtās zināšanas ir tiešām jaunas un saistošas, nozares ekspertu veikto rezultātu interpretāciju un iegūto zināšanu ietekmes pārbaudi. Solī zināšanu izguves procesu var pārskatīt, lai noteiktu, kuras alternatīvās darbības varēja veikt, lai uzlabotu rezultātus. Tiek sagatavots procesā pieļauto kļūdu saraksts;
6. iegūto zināšanu pielietošana. Solis sastāv no plānošanas, kur un kā atklātās zināšanas tiks izmantotas. Paredz ieviest iegūto zināšanu īstenošanas uzraudzības plānu un dokumentēt visu projektu.

1.3.4. STARPINDUSTRIJU STANDARTA PROCESA MODELIS (CRISP-DM)

CRISP-DM ir nozares un instrumentu neitrāls procesa modelis, ko 1996. gada beigās izstrādāja trīs tobrīd tirgus līderi: Daimler, SPSS un NCR (Osei-Bryson & Barclay, 2015). 1997. gadā tika izveidots konsorcijs, kura mērķis bija procesa modeļa veidā formalizēt dažādu reālās pasaules organizāciju pieredzi, kuras nodarbojās ar zināšanu izguvi. Viena no šī projekta galvenajām iezīmēm bija koncentrēšanās uz nepatentēta un brīvi pieejama modeļa izveidi, kas palīdzētu zināšanu izguves projektu izpildē.

CRISP-DM apraksta zināšanu izguves projekta dzīves ciklu sešu posmu veidā - biznesa mērķu izpratne, datu izpratne, datu sagatavošana, modelēšana, novērtēšana un izvietošana (skat., 1.7.att.) (Chapman *et al.*, 2000).



1.7. attēls. CRISP-DM procesa modelis (adaptēts no Chapman et al., 2000).

CRISP-DM ir aprakstīti uzdevumi un darbības, kas jāveic katrā no šīm procesa fāzēm (Chapman et al., 2000):

1. biznesa izpratne. Pirmais solis koncentrēts uz izpratnes iegūvi un projekta mērķu un prasību definēšanu vadoties no biznesa perspektīvas, un šo prasību pārveidošanas zināšanu izguves problēmas definīcijā, kā arī sākotnējā plāna izstrādē, kas veidots biznesa mērķu sasniegšanai;
2. datu izpratne. Iekļauj datu vākšanu, datu analīzi, datu kvalitātes problēmu identificēšanu, ieskatu atklāšanu datos, interesantu apakškopu identificēšanu, ar nolūku veidot hipotēzes par slēpto informāciju;
3. datu sagatavošana. Ietver darbības, lai no sākotnēji neapstrādātajiem datiem izveidotu izmantojamu datu kopu (datus, kas tiks pielietoti modelēšanas rīkā). Datu sagatavošanas uzdevumi tiek izpildīti vairākas reizes, bez noteiktas secības. Sagatavošanas uzdevumos ietilpst tabulu, ierakstu un atribūtu atlase, kā arī datu transformācija un tīrīšana;
4. modelēšana. Tiek noteikti un pielietoti dažādi modeļi, un to parametri tiek kalibrēti lai sasniegtu optimālas vērtībām. CRISP-DM norāda, ka vienam un tam pašam datu izguves problēmas veidam parasti ir izmantojamas vairākas

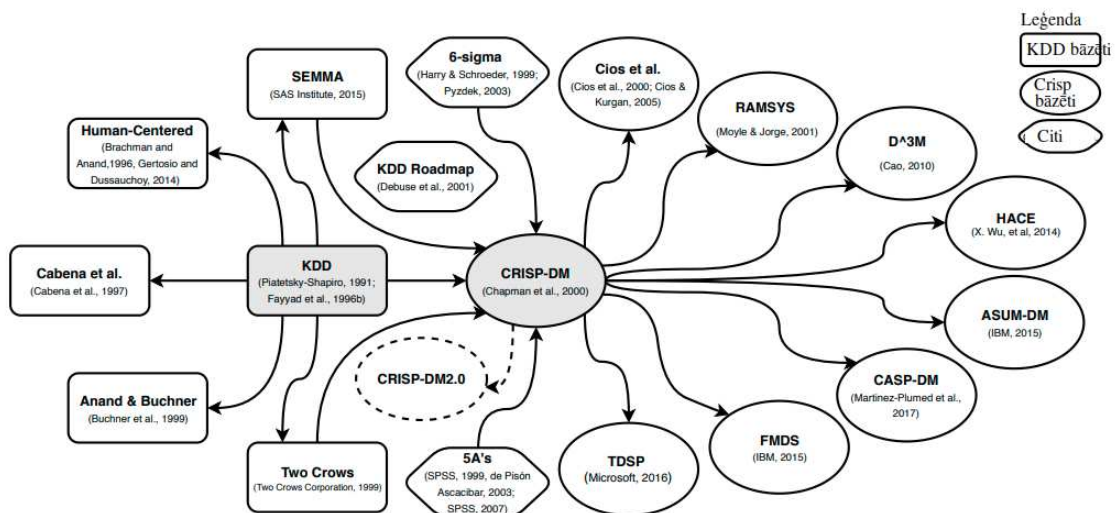
metodes. Dažām metodēm ir īpašas prasības attiecībā uz datu formu, tādēļ bieži ir nepieciešams atgriezties datu sagatavošanas posmā;

5. novērtēšana. Iekļauj rūpīgu modeļa novērtēšanas un izpildīto darbību pārskatīšanu, lai pārlicinātos, ka tas pareizi sasniedz biznesa mērķus. Būtiskākais uzdevums ir identificēt vai kāds nozīmīgs uzņēmējdarbības jautājums nav pietiekami apsvērts;
6. izvietošana. CRISP-DM modeļa izveide nav projekta noslēgums. Ja projekta mērķis ir palielināt zināšanas par noteiktu problēmu, tad iegūtos rezultātus ir nepieciešams arī organizēt un prezentēt tā, lai gala lietotājs tās varētu izmantot. Saskaņā ar CRISP-DM procesa modeli, atkarībā no prasībām, izvietošanas fāze var būt tik vienkārša kā atskaišu ģenerēšana vai tik sarežģīta kā atkārtojama zināšanu izguves procesa ieviešana visā uzņēmumā.

CRISP-DM modelis iekļauj arī vairākas atgriezeniskās saites cilpas, lai uzsvērtu, ka daži procesa soļi ir jāpārskata, lai izmantotu jauno informāciju vai zināšanas, kas iegūtas nākamajā solī (Chapman et al., 2000). Atgriezeniskās cilpas uzrādītas arī 1.6. attēlā, piemēram, lai gan datu sagatavošana parasti notiek pirms modelēšanas, var būt nepieciešams atkārtoti pārskatīt datu sagatavošanu, jo izvēlētajai modelēšanas metodei var būt nepieciešams datus sagatavot citā veidā.

1.3.5. MODEĻU ATTĪSTĪBA

Pirmie zināšanu izguves procesu modeļi tika izstrādāti 90. gadu beigās. Balstoties uz vairākām aptaujām pētnieki F. Martínez-Plumed et al. (Martinez-Plumed et al., 2021) un Rotondo A., Fergus Q. (Rotondo & Quilligan, 2020) apgalvo, ka CRISP-DM joprojām ir noklusētais standarts datu ieguves un zināšanu izguves projektu izstrādei. Gandrīz trīsdesmit gadu laikā nozare ir ievērojami attīstījusies un datu zinātne tagad ir vadošais termins. Zināšanu izguves procesu modeļu attīstība un metodoloģijas ir attēlotas 1.8.attēlā.



1.8.attēls. Zināmākie zināšanu izguves modeļi (adaptēts no Martinez-Plumed et al., 2021).

Attēlā redzamās bultiņas (skat., 1.8.att.) norāda, ka CRISP-DM ietver principus un idejas no vairākām metodoloģijām, vienlaikus veidojot pamatu vēlāk izstrādātiem procesu modeļiem. Vairāki jauni zināšanu izguves procesu modeļi ir attīstījušies kā CRISP-DM paplašinājumi, vienlaikus demonstrējot, kā modeļi var modernizēt, to būtiski nemainot:

- RAMSYS (angliski: RApid Cooperative Data Mining SYStem) (Moyle & Jorge, 2001), metodika projektu izstrādei ar ģeogrāfiski atšķirīgām grupām;
- ASUM-DM (angliski: Analytics Solutions Unified Method for Data Mining/Predictive Analytics) (IBM, 2005), kas uzlabo un paplašina CRISP-DM, pievienojot infrastruktūras, darbību, izvietojanas un projektu pārvaldības sadaļas, kā arī veidnes un vadlīnijas;
- CASP-DM (Martínez-Plumed et al., 2017), risina specifiskas mašīnmācīšanās un datu ieguves problēmas konteksta maiņai un modeļu atkārtotas izmantošanas apstrādei;
- HACE (Xindong Wu et al., 2014), lielo datu apstrādes sistēma, kuras pamatā ir trīs līmeņu struktūra: lielo datu ieguves platforma (I līmenis), problēmas saistībā ar informācijas apmaiņu un privātumu, kā arī lielo datu lietojumprogrammu domēni (II līmenis) un lielo datu ieguves algoritmi (III līmenis).

Lielais datu apjoms, kā arī datu zinātnes projektu eksperimentālais un izpētes raksturs, prasa mazāk definētas, vieglākas un elastīgākas metodoloģijas. Vairāki IT uzņēmumi ir ieviesuši jaunas, elastīgākas metodoloģijas datu zinātnes projektiem,

piemēram, IBM FMDS (angliski: Foundational Methodology for Data Science) (Rollins, 2015) un Microsoft TDSP (angliski: Team Data Science Process) (Severtson, 2021).

Zināšanu izguves process ir būtiski mainījies kopš CRISP-DM procesa izveides, būtiska joma, kurā CRISP-DM modelis nestrādā pietiekami korekti, ir “uz datiem balstīti produkti”, piemēram, mobilā lietotne, kas iegūst informāciju no lietotāju atrašanās vietas un iesaka maršrutus citiem lietotājiem atbilstoši viņu vajadzībām. Produkts ir dati un no tiem iegūtās zināšanas (Martinez-Plumed et al., 2021). Datiem var būt vairāki izmantošanas veidi arī sfērā, kas nav saistīta ar domēnu, kurā dati tika iegūti. Tā, piemēram, savāktie dati no elektronisko norēķinu sistēmu lietošanas var tikt pārdoti un izmantoti uzņēmumos, lai noskaidrotu piemērotāko vietu jaunam veikalam. Datu apjoms un sarežģītība lietojumprogrammās mūsdienās liecina arī par to, ka datu apstrādei ir nepieciešams nozīmīgs tehnisks darbs pie pārvaldības un infrastruktūras. Savukārt, CRISP-DM modelī “dati” iekļauti kā statistiska vienība procesa vidū (skat. 1.6. att.) (Martinez-Plumed et al., 2021), kas nozīmē, ka zināšanu izguves procesu ir nepieciešams aplūkot kontekstā ar zināšanu izguves ietvaru, kurā process tiek pielietots.

1.4. ZINĀŠANU IZGUVES IETVARI

Neskatoties uz pētnieku piedāvātajiem procesu modeļiem zināšanu izguves sfērā, veiksmīga ieviešana uzņēmuma ikdienā saskaras ar izaicinājumiem. Lielākā daļa datu izguves algoritmu un rīku apstājas pie tādu modeļu izveides un piegādes, kas apmierina primāri tehniskās prasības. Situācijās, kad zināšanu izguves modeļi tiek izstrādāti, uzņēmuma iekšienē ne vienmēr ir zināšanas, izpratne un interese kā izstrādātos modeļus adaptēt un izmantot biznesa lēmumu pieņemšanā (Cao et al., 2010). Praktisko zināšanu izguve ir slēgts problēmu risināšanas process, kas ietver virkni mērķtiecīgu darbību: problēmas definēšanu, ietvara/modeļa izstrādi. Process paredzēts, lai nodrošinātu funkcionējošus biznesa noteikumus, kurus var saistīt vai integrēt ar biznesa procesiem un sistēmām. Šāda un līdzīga veida problēmas ir ietekmējušas vērienu kādā tiek izmantotas datu izguves metodes (Cao et al., 2010).

Neskatoties uz augstāk minētajiem izaicinājumiem, IT sistēmās ir vērojama pieaugoša tendence informācijas un slēpto korelāciju iegūšanai izmantot produktam pieejamos datus. Būtiskākie faktori, kas to veicinājuši ir vēlme sniegt labākus pakalpojumus lietotājiem un lēmumu pieņemšanas procesu uzlabošana. Un, lai tas tiktu

sasniegts, tiek integrētas un analizētas ģenerētās datu kopas no dažādiem domēniem (Osman, 2019).

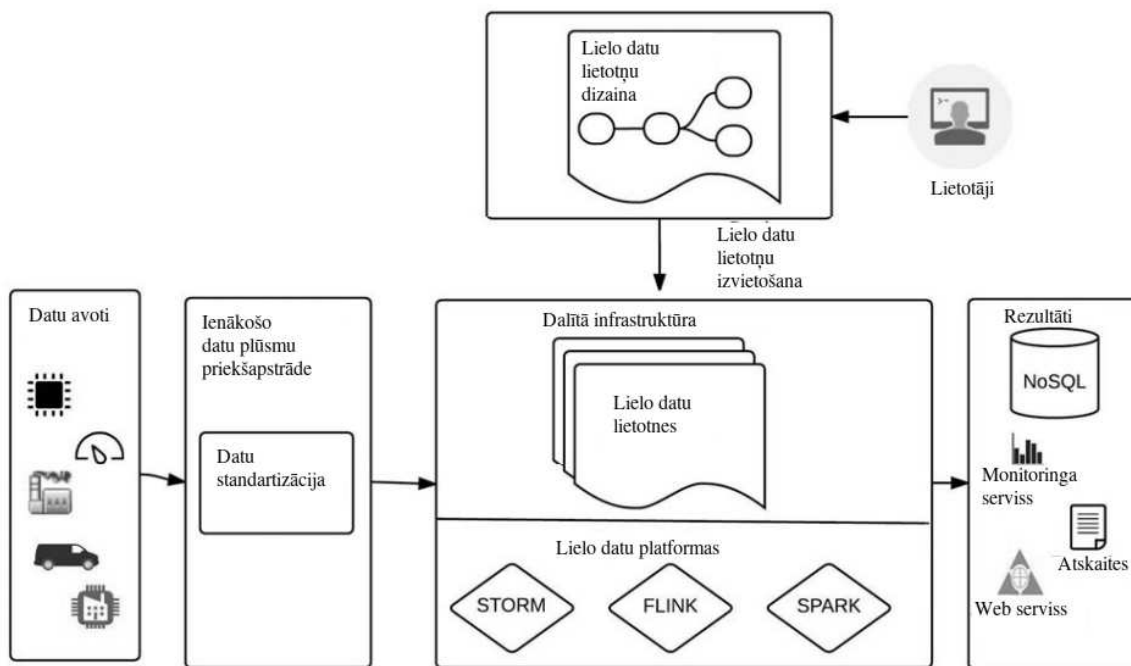
Darba daļā tiek aplūkoti vairāki zināšanu izguves ietvari. Zināšanu izguves ietvari izvēlēti tā, lai katrs no tiem aplūkotu citu sfēru, kā arī, lai tie būtu potenciāli inovatīvi un to iepazīstinošās publikācijas būtu izmantotas saistītos pētījumos. Lai nodrošinātu atlasī, noteikti sekojoši kritēriji:

- zināšanu izguves ietvars nevar būt vecāks par pieciem gadiem (ietvari izvēlēti 2021. gadā);
- ietvaru pārstāvošā publikācija, atbilstoši, Scopus svērtajai citātu ietekmei (angliski: Field-Weighted Citation Impact) (Scopus, 2020) novērtēta ar vērtību vismaz deviņi, kas nozīmē, ka šī publikācija ir nozīmīga izvēlētajā nozarē.

Balstoties uz izvirzītajiem kritērijiem aplūkoti trīs ietvari. Inovatīvs lielo datu analīzes ietvars viedajām pilsētām (Osman, 2019). Kognitīvā skaitļošana: arhitektūra, tehnoloģijas un viedās lietojumprogrammas (Chen et al., 2018). Lielie dati nozarei 4.0: konceptuāls ietvars (Gokalp et al., 2016).

1.4.1. KONCEPTUĀLS IETVARS INDUSTRIJAI 4.0

Gokalps (Gokalp) uzskata, ka lielo datu analīzes platformu izmantošanai un uzstādīšanai ir nepieciešamas ievērojamas zināšanas datu zinātnē un IT jomā kopumā, to sarežģītās infrastruktūras un programmēšanas modeļu ietekmē. Pretējā gadījumā tas varētu kavēt lielo datu tehnoloģiju ieviešanu nozarē 4.0. Tādējādi, no lietotāju spējas pieņemt, lielo datu platformām būtu nepieciešams tāds programmēšanas modelis, kas nodrošinātu augstāka līmeņa abstrakcijas. Šajā ietvarā tiek piedāvāts uz vizuālu un datu plūsmu balstīts arhitektūras ietvars, kas abstrahētu izstrādātājus no datu apstrādes platformu sarežģītības (Gokalp et al., 2016). Gokalpa piedāvātais konceptuālais ietvars aplūkojams 1.9.attēlā, tas iekļauj sekojošus moduļus - lielo datu lietotnes modulis, ieejas datu priekšapstrāde, dalīta infrastruktūra un rezultātu izplatīšana (skat., 1.9.att.).

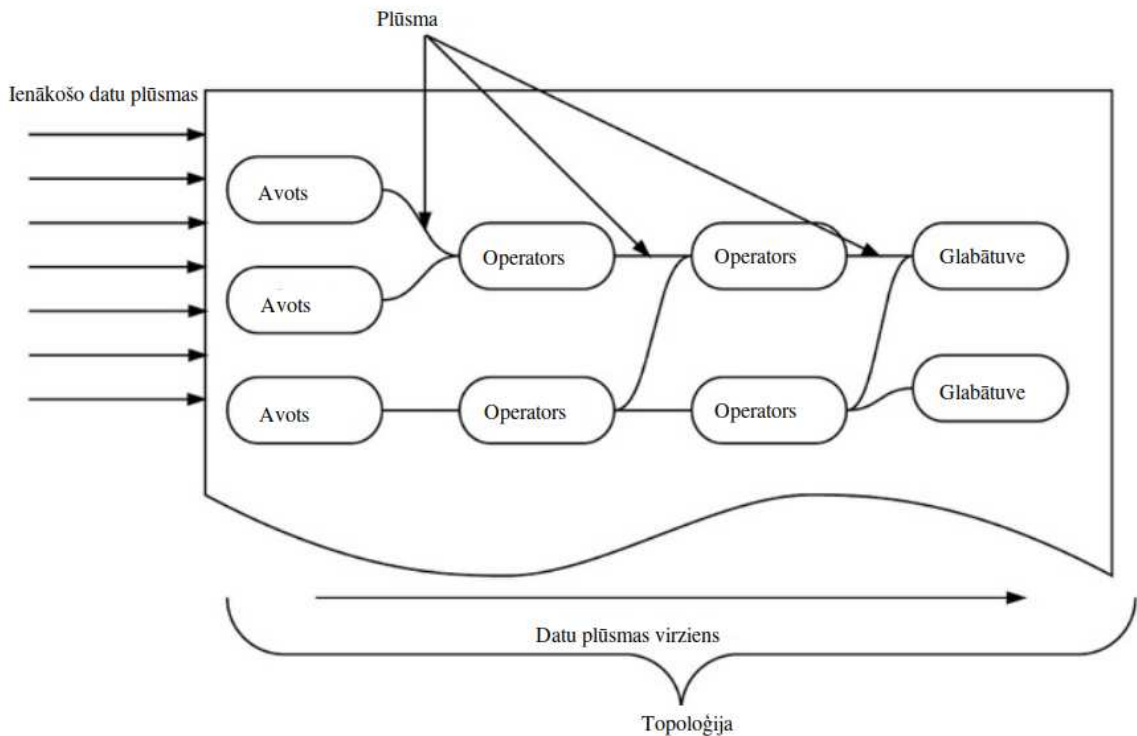


1.9.attēls. Gokalp et al., piedāvātais konceptuālais ietvars (adaptēts no Gokalp et al., 2016).

Gokalpa piedāvātajā ietvarā, lietojumprogrammu izstrādes modulis ļauj sistēmu inženieriem izstrādāt brīvi izvēlētas lietojumprogrammas. Lietojumprogrammas tiek attēlotas kā virzīti grafiki, kur virsotnes attēlo datu ieguves un mašīnmācīšanās algoritmus, kā arī programmēšanas konstrukcijas, savukārt malas apzīmē datu plūsmas, kas atbilst starprezultātiem (skat., 1.10.att.). Programmēšanas mezgli var pievienot un apstrādāt datus vienotā standartā no dažādiem avotiem un tikt integrēti ar citiem programmēšanas mezgliem. Lietojumprogrammu loģiku var izveidot savienojot programmēšanas mezglus, neuztraucoties par to iekšējām darbībām un saskarnēm (Gokalp et al., 2016).

Ieejas dati Gokalpa piedāvātajā ietvarā tiek ģenerēti atšķirīgos formātos, tādēļ datu dažādība ir nozīmīgs izaicinājums. Ieejas datu priekšapstrādes modulim ir nozīmīga loma - nodrošināt datu pārveidošanu vienotā formātā turpmākai apstrādei (Gokalp et al., 2016). Ietvara lietojumprogrammām savukārt nepieciešama augstas veiktspējas un mērogojama infrastruktūra, kas rada nepieciešamību pēc dalītas (angliski: distributed) infrastruktūras. Lielo datu apstrādes lietojumprogrammu prasības atšķiras atkarībā no lietošanas gadījumiem, tādējādi neeksistē vienots lielo datu risinājums visām situācijām (Gokalp et al., 2016). Piedāvātā ietvara nolūks ir atbalstīt vairākas lielo datu platformas, piemēram, Flink, Spark un Storm (Apache Flink, 2021; Apache Spark, 2021; Apache Storm, 2021).

Atbilstoši izstrādātās lietojumprogrammas specifiskajām īpašībām var izvēlēties vienu no atbalstītajām platformām (Gokalp et al., 2016). Ņemot vērā izstrādāto lietojumprogrammu loģiku un lietošanas gadījumus, ietvars var piedāvāt piemērotāko lielo datu platformu lietojumprogrammu izstrādei.



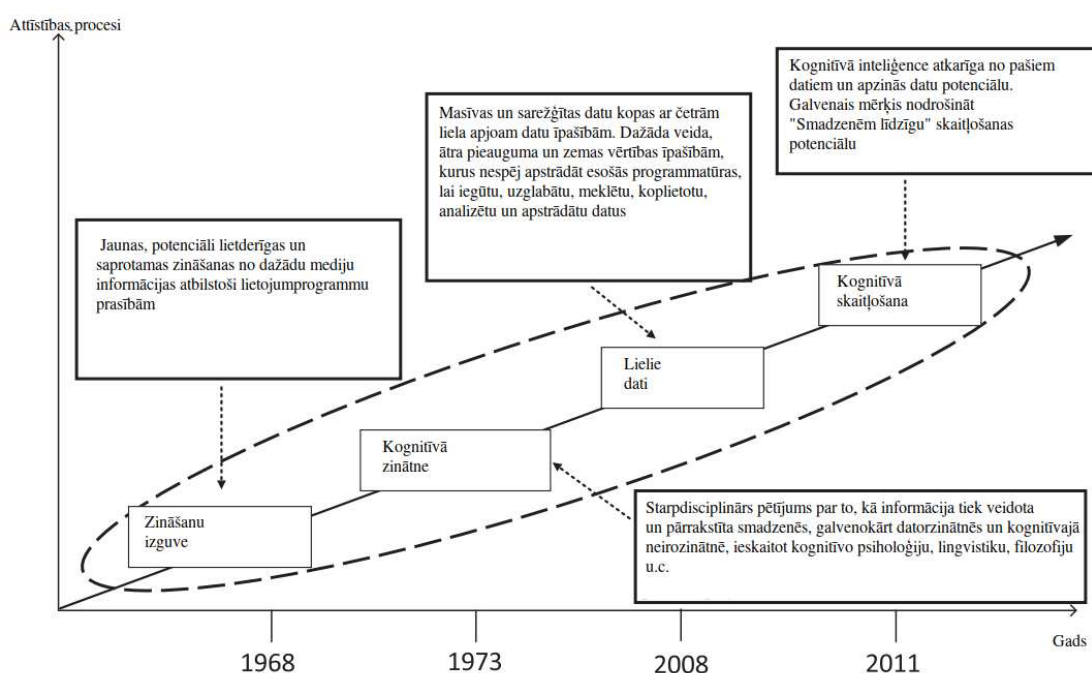
1.10.attēls. Programmēšanas mezgli Gokalpa ietvarā (adaptēts no Gokalp et al., 2016).

Saskaņā ar Gokalpu iegūtie rezultāti var tikt nosūtīti ieinteresētajām pusēm vairākos veidos. Katrs izplatīšanas kanāls piedāvātajā ietvarā tiek definēts kā programmēšanas mezgls. Gala lietotāji var izvēlēties vairāk nekā vienu izplatīšanas kanālu rezultātu iegūšanai. Ietvars neizslēdz iespēju piegādāt rezultātus arī ārējām vienībām, izmantojot tīmekļa pakalpojumus datu vizualizācijas vai uzraudzības nolūkos (Gokalp et al., 2016).

Piedāvātais konceptuālais ietvars uzrāda potenciālās zināšanu izguves sistēmas komponentes, tomēr tajā netiek detalizēti moduļu darbības pamatprincipi, konkrētais ietvars strikti saglabā konceptuālo pieeju.

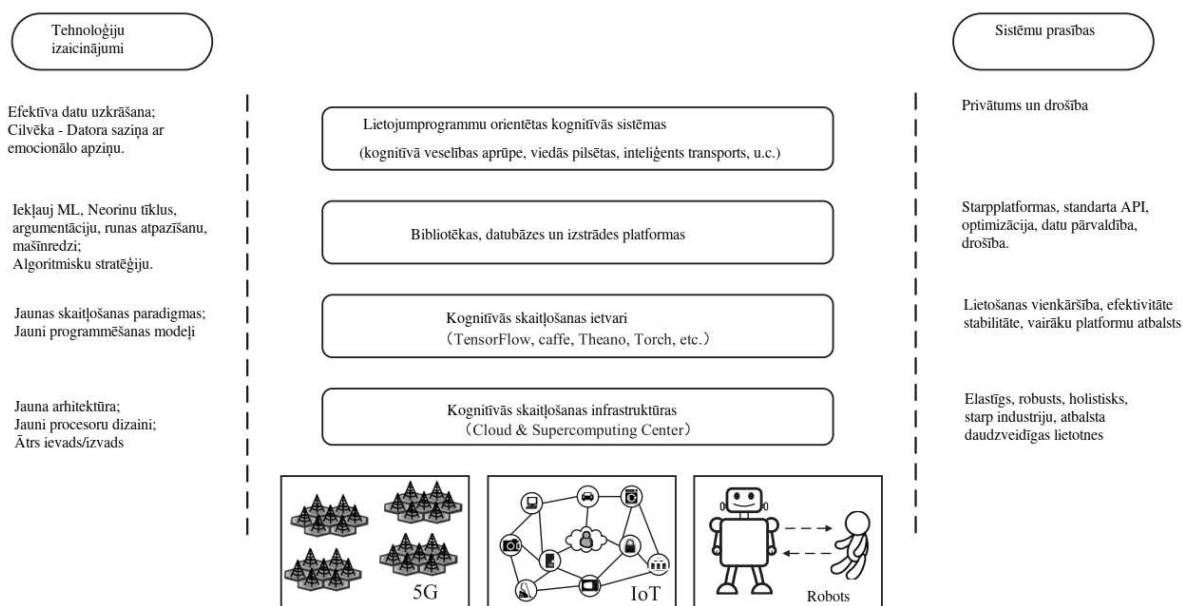
1.4.2. KOGNITĪVĀ SKAITĻOŠANA: ARHITEKTŪRA, TEHNOĻIJKAS UN VIEDĀS LIETOTNES

Čens (Chen) piedāvā kognitīvo skaitļošanu ierindot mākslīgā intelekta apakšnozarē, kura precīzāk simulē cilvēka domāšanas procesu, izmantojot pašmācības algoritmus, zināšanu izgūvi, paraugu atpazīšanu un dabiskās valodas apstrādi. Šīs mākslīgās vides paļaujas uz dziļās mācīšanās algoritmiem un neironu tīkliem, lai apstrādātu informāciju, to salīdzinot ar mācību datu kopu. Atdarīnot cilvēka domāšanas procesu, datori palīdz indivīdiem pieņemt korektākus lēmumus. Ņemot vērā iekārtu un cilvēka mijiedarbību, kognitīvo skaitļošanu var raksturot arī kā paplašināto intelektu (Chen et al., 2018).



1.11.attēls. Kognitīvās skaitļošanas attīstība (adaptēts no Chen et al., 2018).

Attēlā (skat., 1.11.att.) uzrādīts kognitīvās skaitļošanas attīstības process, kas iekļauj gan zināšanu izgūvi, gan kognitīvās zinātnes, gan lielos datus. Kognitīvā skaitļošana koncentrējas uz atklājumiem datu apstrādes metodēs un tajā apstrādātie dati ne vienmēr ir lieli dati (Chen et al., 2018).



1.12.attēls. Kognitīvās skaitļošanas ietvara arhitektūra (adaptēts no Chen et al., 2018).

Čens uzskata, ka ar tādu tehnoloģiju atbalstu kā 5G, robotika, padziļināta mācīšanās, lietu internets, mākoņskaitļošanas infrastruktūra, var tikt risināti izaicinājumi, kuri saistīti ar cilvēka un iekārtas mijiedarbību, bals atpazīšanu un datora redzi. Potenciāli atbalstītās pielietojuma sfēras var būt veselības uzraudzība, kognitīvā veselības aprūpe, viedā pilsēta, viedais transports un zinātniskie eksperimenti. Katrs ietvara arhitektūras slānis (skat., 1.12.att.) ir saistīts ar atbilstošiem tehnoloģiskiem izaicinājumiem un ietvara tehniskajām prasībām (Chen et al., 2018).

Piedāvātajā ietvarā (skat., 1.12.att.) lietu internets iegūst informāciju par objektiem, izmantojot, piemēram, RFID un bezvadu sensorus, satelīta un WiFi pozicionēšanu un pirkstu nospiedumus. Izmantojot saziņas līdzekļus, lietu internets nodrošina informācijas izplatību, veic koplietošanu un integrāciju kā arī informācijas analīzi un apstrādi. Lēmumu pieņemšanas procesiem tiek izmantotas viedās skaitļošanas tehnoloģijas, piemēram, mākoņdatošana, mašīnmācīšanās un zināšanu izguve. Savukārt kognitīvā skaitļošana spētu nodrošināt rīkus ar labāku energoefektivitāti datu uztverei un ieguvei no lietu interneta (Chen et al., 2018).

Viena no atšķirībām starp lielo datu analīzi un kognitīvo skaitļošanu ir datu apjoms. Lielo datu analīze, pēc pētnieka Čena, ne vienmēr ir uzskatāma par kognitīvo skaitļošanu. Lielajiem datiem galvenā uzmanība tiek pievērsta zināšanu izguvei no liela datu apjoma, savukārt bez liela datu apjoma nav iespējams garantēt prognozes precizitāti un ticamību.

Kognitīvā skaitļošana savukārt nepaļaujas uz lielu datu apjomu, bet balstās uz izziņu un spriedumu, līdzīgi kā cilvēka smadzenes.

Čens uzskata, ka mākoņskaitļošana virtualizē skaitļošanu, datu krātuves un komunikāciju joslas, tādējādi tā samazina programmatūras pakalpojumu izvietojuma izmaksas un nodrošina atbalstu industrializācijai, kā arī lietojumprogrammas kognitīvās skaitļošanas veicināšanai. Turklāt mākoņskaitļošanas jaudas nodrošina dinamiskus, elastīgus, virtuālus, koplietotus un efektīvus skaitļošanas resursu pakalpojumus kognitīvajiem skaitļošanas risinājumiem (Chen et al., 2018).

Lai arī pētnieks Čens iepazīstina ar potenciālo ietvara arhitektūru, netiek sniegti skaidrojumi nedz par potenciāli pielietojamajām tehnoloģijām ietvaros, nedz par to izvietojumiem, datu glabātuvēm un starpmoduļu saziņu. Pētnieku piedāvātais ietvars saglabā konceptuālo līmeni nesniedzot detalizētāku skaidrojumu potenciālajām pielietojuma sfērām.

1.4.3. INOVATĪVS LIELO DATU ANALĪZES IETVARS VIEDAJĀM PILSĒTĀM

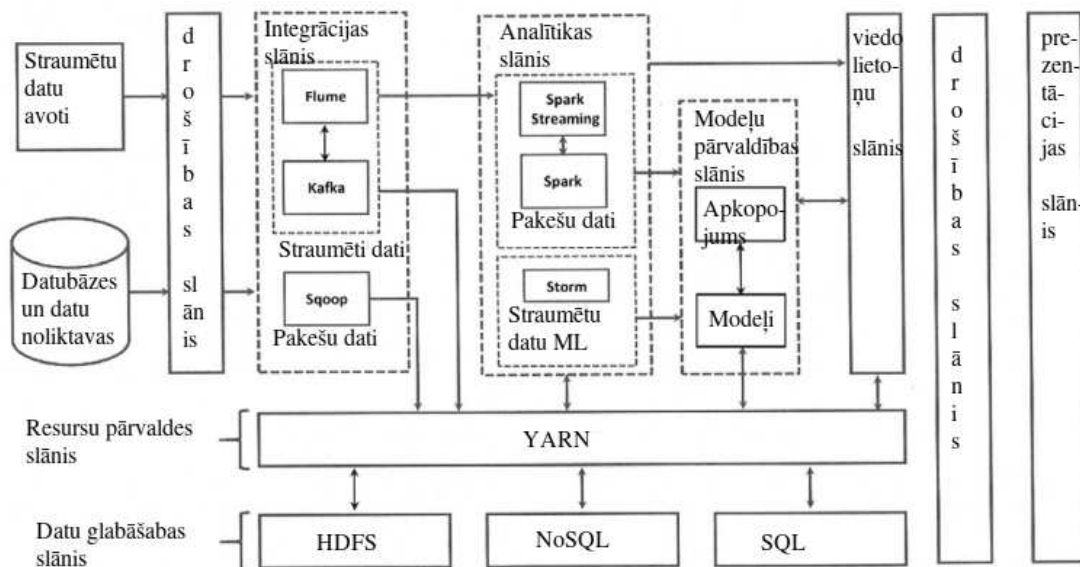
Zināšanu izguvei un slēpto korelāciju analīzei no datiem, lai sniegtu labākus pakalpojumus gala lietotājiem un atbalstītu lēmumu pieņemšanas procesu, ir pieaugoša tendence informācijas sistēmās. Pētnieks Osmans (Osman, 2019) uzskata, lai iegūtu vērtīgu ieskatu pilsētas līmeņa viedo informācijas pakalpojumu izstrādē, ģenerētās datu kopas no dažādiem pilsētas domēniem ir jāintegrē un jāanalizē un šo procesu sauc par lielo datu analīzi vai lielo datu vērtību ķēdi. Publikācijā Inovatīvs lielo datu analīzes ietvars viedajām pilsētām (angliski: “A novel big data analytics framework for smart cities”) tiek uzrādīts ietvars viedajām pilsētām ar nosaukumu “Viedās pilsētas datu analīzes panelis - SCDAP” (angliski: “Smart City Data Analytics Panel — SCDAP”) (Osman, 2019).

Piedāvātā SCDAP ietvara konceptuālā arhitektūra redzama 1.13. attēlā. To veido trīs slāņu arhitektūra: platformas slānis, drošības slānis un datu apstrādes slānis (Osman, 2019). Atbilstoši pētnieka Osmana sniegtajam skaidrojumam, ietvara slāņiem ir sekojoša funkcionalitāte:

- platformas slānis - horizontāli mērogojams, ietver aparatūras kopas, operētājsistēmas un sakaru protokolus. Horizontāli mērogojamās platformās pēc vajadzības iespējams pievienot papildu skaitļošanas jaudas;
- drošības slānis - funkcionalitātes prasības skaidri definējamas tikai fiziskas ieviešanas laikā. Projektēšanā jāievēro šādi drošības pasākumi - ierobežota

4. reāllaika analītika - nodrošina straumētu datu apstrādi tuvināti reālajam laikam. Reāllaika lietojumprogrammas ir tādas, kas darbojas laika posmā, ko gala lietotājs uztver kā tūlītēju;
5. pakešu datu krātuve - uzglabāšanas pārvaldības sistēma (Hadoop HDFS) (Apache Hadoop, 2021), NoSQL (MongoDB Inc, 2021) datu bāzes pārvaldības sistēmas;
6. pakešu datu analīze - lietojumprogrammām, kuras var darboties ar apstrādes aizkavi;
7. modeļu vadība - datu modeļu pārvaldības komponente, kurā izveidotos datu analīzes modeļus var uzglabāt, izgūt vai dzēst kopā ar atbilstošajiem metadatiem turpmākiem pieprasījumiem vai atkārtotai izmantošanai;
8. modeļu apkopošana - datu modeļu ansambļa funkcionalitāte augstāka līmeņa un sarežģītības analīzei un pieprasījumiem;
9. viedās lietotnes - balstītas uz iegūtajiem datu analīzes modeļiem;
10. lietotāju saskarne - lai nodrošinātu efektīvus, elastīgus rīkus, kas nodrošina piekļuvi, ziņošanu un pieprasījumus.

Osmana (Osman, 2019) piedāvā konceptuālā SCDAP ietvara aprobāciju izmantojot divas pieejas - Apache Hadoop (Apache Hadoop, 2021) (skat., 1.14.att.) un TensorFlow (TensorFlow, 2022), Apache Hadoop un Anaconda (Anaconda, 2022).



1.14.attēls. SCDAP ietvara aprobācija ar Apache Hadoop (adaptēts no Osman, 2019).

SCDAP aprobācija pielietojot Apache Hadoop redzama 1.13. attēlā, pētnieks norāda, ka Apache Hadoop ir horizontāli mērogojama platforma lielu datu glabāšanai, pārvaldībai un apstrādei, kas atbilst viedās pilsētas lielo datu analītikas lietotnēm. Piedāvāto arhitektūru veido astoņi slāņi (skat., 1.14.att.):

1. resursu vadības slānis, pielietojot YARN (Yet, Another Resource Negotiator) (*Yarn*, 2022). YARN izolētu resursu pārvaldību un plānošanu no datu apstrādes komponentēm;
2. datu glabāšanas slānis - Hadoop HDFS vai NoSQL datubāzes kā HBase (*HBase*, 2022) vai Kudu (*Kudu*, 2022) kā arī alternatīva SQL datubāzes;
3. datu integrācijas slānis - piedāvātie tehnoloģiskie risinājumi Apache Flume (*Apache Flume*, 2022), Apache Kafka (*Apache Kafka*, 2022) un Apache Sqoop (*Apache Sqoop*, 2022);
4. datu analīzes slānis - piedāvātās tehnoloģijas ir Apache Spark (*Apache Spark*, 2021) un Apache Storm (*Apache Storm*, 2021);
5. modeļu pārvaldības slānis - specifiskas tehnoloģijas netiek identificētas;
6. viedo lietotņu slānis - slānis, kurā iespējams pievienot vai izstrādāt individuālas un pielāgotas lietotnes;
7. drošības slānis - tiek piedāvāts Sentry (*Apache Sentry*, 2021) kā sistēma uz lomu balstītas autorizācijas ieviešanai datiem, kas tiek glabāti Hadoop klāsteros. Jāmin, ka Sentry projekts tika pārtraukts 2020 gada decembrī;
8. lietotāju saskarnes slānis - iegūtās zināšanas un modeļus var vizualizēt ar vienkāršām, interaktīvām un efektīvām grafiskajām lietotāju saskarnēm (GUI), specifiski risinājumi netiek piedāvāti, tie atkarīgi no individuālajām prasībām.

Kā alternatīvas tehnoloģijas, kuras var izmantot SCDAP ietvara izveidei tiek minētas atvērtā koda tehnoloģijas Anaconda un TensorFlow, kā arī komerciālās platformas SAS un RapidMiner (Osman, 2019). SCDAP ietvara izstrādei tiek pielietota DSRM (angliski: Design Science Research Methodology) (Peffer et al., 2007) metodoloģija. SCDAP tiek izvirzīti seši nosacījumi - slāņu pieeja, standartizēta datu ieguve un izmantošana, reāllaika un vēsturisko datu analītika, iteratīva un secīga datu apstrāde, modeļu pārvaldība, modeļu apkopošana (Osman, 2019).

Osman darbā piedāvātais ietvars tiek apskatīts viedo pilsētu kontekstā, aplūkojot potenciālās ietvaram izvirzāmās funkcionālās un drošības prasības. Piedāvātais SCDAP

ietvars iekļauj ne tikai kontekstuālo modeli, bet arī potenciālu implementāciju ar noteiktu tehnoloģiju pielietojumu.

1.4.4. IETVARU SALĪDZINĀJUMS

Salīdzinot novērtētos ietvarus (skat., 1.3.tabula) izvirzītas pazīmes, kuras pētnieki uzrādījuši izveidotajiem zināšanu izguves ietvariem. Apkopotas tās pazīmes, kuras uzrādījuši pētnieki savās publikācijās. Izvirzītās pazīmes netiek uzskatītas par galīgu zināšanu ietvaru raksturojošo elementu kopu, jo šie ietvari aptver plašu spektru.

1.3. tabula. Ietvaru salīdzinājums (autora veidota)

Pazīme	Ietvars industrijai 4.0 (skat., 1.4.1. nodaļu)	Kognitīvās skaitļošanas ietvars (skat., 1.4.2. nodaļu)	Ietvars viedajām pilsētām (skat., 1.4.3. nodaļu)
Piedāvā implementācijas aprakstu	X	X	✓
Modulārs	✓	✓	✓
Izstrādātāji tiek abstrahēti no datu apstrādes platformu sarežģītības	✓	X	✓
Dalīta infrastruktūra	✓	X	✓
Datu priekšapstrādes modulis	✓	X	✓
Rezultātu apstrādes un izplatīšanas modulis	✓	✓	✓
Savienojami programmēšanas mezgli	✓	X	X
Augstas veiktspējas mērogojama infrastruktūra	✓	X	✓
Lielo datu platformu atbalsts	✓	X	✓
Uzrādītas potenciālās tehnoloģijas un to pielietojums	X	X	✓

Pazīme	Ietvars industrijai 4.0 (skat., 1.4.1. nodaļu)	Kognitīvās skaitļošanas ietvars (skat., 1.4.2. nodaļu)	Ietvars viedajām pilsētām (skat., 1.4.3. nodaļu)
Tiešsaistes analītikas modulis	X	X	✓
Mērķauditorija	✓ (rūpniecība)	✓ (lietu internets)	✓ (viedās pilsētas)
Drošība un tiesību aktu ierobežojumi kā daļa no ietvara	X	X	X

Plašākā informāciju par zināšanu izguves ietvaru uzrādīta viedo pilsētu inovatīvajam risinājumam (skat. 1.3.tabula), kur apskatīti potenciālie aprobācijas scenāriji. Apskatītie zināšanu ietvari piedāvā potenciāli inovatīvas pieejas zināšanu izguves procesu izveidei, tomēr tie neaplūko detalizēti nedz mērogojamās tehnoloģijas, nedz sistēmu arhitektūru, nedz datu ieguves un izplatīšanas scenārijus, nedz arī uzrāda ietvaru aprobāciju noteiktā nozarē, kas rada nepieciešamību analizēt zināšanu izguves ietvaru tehnoloģiskos apsvērumus.

1.5. NODAĻAS KOPSAVILKUMS UN SECINĀJUMI

Nonaka un Takeuchi (1995) uzskata, ka zināšanu pamatojuma veids saistāms ar sociālo kontekstu, kurā zināšanas tiek radītas. Praksē tas nozīmē, ka jaunu zināšanu radīšana jāvērtē nevis ar loģikas, bet ar lietderības metriku.

Datu apjoms un sarežģītība nosaka, ka datu apstrādei nepieciešams nozīmīgs tehnisks darbs ar pārvaldību un infrastruktūru, savukārt CRISP-DM modelī dati iekļauti kā statistiska vienība, kas nozīmē, ka zināšanu izguves procesu nepieciešams aplūkot kontekstā ar zināšanu izguves ietvaru, kurā process tiek pielietots.

Gokalp et al. (2016), uzskata, ka lielo datu analīzes platformu izmantošanai un uzstādīšanai nepieciešamas ievērojamas zināšanas datu zinātnes un IT jomā to sarežģītās infrastruktūras un programmēšanas modeļu ietekmes rezultātā. No lietotāju pieņemšanas skatupunkta būtu nepieciešams programmēšanas modelis lielo datu platformām, kas nodrošinātu augstāka līmeņa abstrakcijas.

Chen et al. (2018) izvirza apgalvojumu, ka mākoņskaitļošana virtualizē datu apstrādes, datu krātuves un komunikāciju joslas, tādējādi samazinot programmatūras pakalpojumu izvietojšanas izmaksas un nodrošinot atbalstu industrializācijai. Kā arī

mākoņskaitļošanas jaudas nodrošina dinamiskus, elastīgus, virtuālus, koplietotus un efektīvus skaitļošanas resursu pakalpojumus kognitīvajiem skaitļošanas risinājumiem.

Darba pirmajā daļā:

- ir apkopoti jēdzieni un definīcijas par zināšanu izguvi, viedajām sistēmām un lietotāju reakcijas datiem, sniedzot ieskatu par minēto jēdzienu izpratni un pielietojumu saistībā ar zināšanu definīciju;
- ir apkopoti zināšanu izguves procesu modeļi, to vēsture, izcelsme un uzrādīta saikne ar zināšanu izguves sistēmām apkopojot informāciju par zināšanu izguves procesu attīstību un pielietojumu;
- ir izveidota saistošo pētījumu atlases pieeja un meklēšanas stratēģija, kura apkopo detalizētu informāciju par promocijas darbā aplūkotajiem jēdzieniem, identificējot nozīmīgākos pētniekus, sfēras un nozīmīgākās publikācijas;
- ir salīdzināti trīs zināšanu izguves ietvari, kuri izvēlēti tā, lai katrs no tiem aplūkotu citu sfēru, kā arī, lai tie būtu potenciāli inovatīvi un to iepazīstinošās publikācijas būtu izmantotas saistītos pētījumos;
- identificēti un analizēti pieejamie zināšanu izguves ietvari un to tehnoloģiju aprobācija;
- ir izveidota zināšanu izguves ietvaru salīdzināšanas pieeja un identificētas nozīmīgākās pazīmes;
- ir apkopota literatūras analīze esošo teorētisko procesu modeļu un tehnoloģisko iespēju apzināšanai un konceptuālā modeļa izstrādei.

Darba pirmās daļas rezultāti apkopoti zinātniskās publikācijās starptautiski citējamās datubāzēs iekļautajos izdevumos (Jansevskis & Osis, 2022, 2023). Darba pirmajā daļā aprakstītie (skat., 1.4.nodaļu) zināšanu ietvari piedāvā potenciāli inovatīvas pieejas zināšanu izguves procesu izveidei, tomēr šie ietvari detalizēti neaplūko nedz mērogojamās tehnoloģijas, nedz sistēmu arhitektūru, nedz datu ieguves un izplatīšanas scenārijus, kā arī tie neuzrāda ietvaru aprobāciju noteiktā nozarē, kas rada nepieciešamību apskatīt zināšanu izguves ietvaru tehnoloģiskos kritērijus.

2. ZINĀŠANU IZGUVES IETVARU TEHNOLOĢISKIE APSVĒRUMI UN KRITĒRIJI

Lielo datu tehnoloģijas tiek pielietotas zināšanu izguves sistēmu izstrādei, datu analīzei un datu koplietošanai un radījušas vērā ņemamus ekonomiskos ieguvumus. Iegūtās zināšanas nodrošina lēmumu pieņemšanas stratēģijas sociālajai un ekonomiskajai attīstībai. Lielo datu pakalpojumu arhitektūra ir jauns pakalpojumu ekonomiskais modelis, kas datus izmanto kā resursu un ielādē un izvelk no dažādiem datu avotiem iegūto informāciju (Wang et al., 2020). Lielo datu pakalpojumu arhitektūra nodrošina pielāgotas datu apstrādes metodes, datu analīzes un vizualizācijas pakalpojumus.

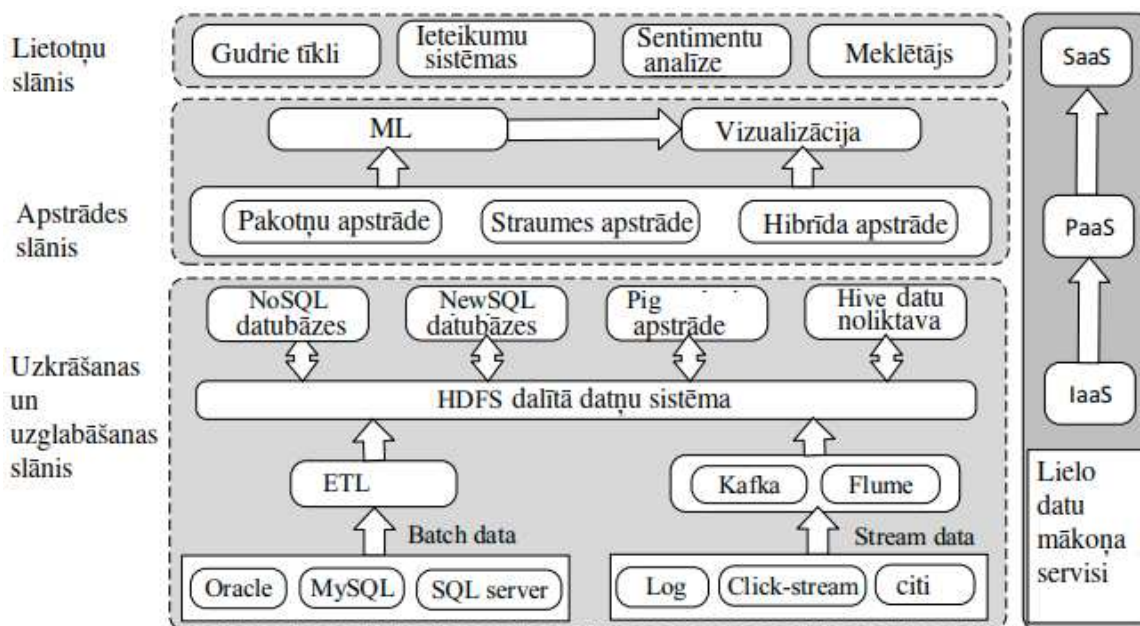
Pēdējās desmitgades laikā ir saražots ievērojams datu daudzums, ko ietekmējis lietu internets (IoT) un iekārtu miniaturizācija. Šādi dati nav lietderīgi bez analītikas iespēju pielietošanas. Vairāki lielo datu, IoT un analītikas risinājumi ir ļāvuši lietotājiem iegūt lietderīgu ieskatu lielajos datos (Mohsen et al., 2017).

Lielie dati un mākoņskaitļošanas tehnoloģijas tiek uzskatīti par galvenajiem faktoriem, kas ietekmēs nākotnes industriālās ekosistēmas veidošanos. Viens no faktoriem ir tas, ka zināšanu izguves sistēmas, izmantojot mākoņskaitļošanas tehnoloģijas, spēj pārvaldīt rūpnieciskās vides radīto datu daudzveidību, ātrumu, apjomu un kritiskumu, izmantojot augsti sadalītas un mērogojamas arhitektūras, lai apstrādātu datus tuvināti reālajam laikam (López Martínez et al., 2021).

Covid-19 pandēmija, karantīna un mājseide uzstādīja mobilo datu patēriņa rekordus. Dažas prakses, kas tika uzskatītas par ārkārtējām, kļuva ļoti izplatītas, piemēram, attālinātais darbs, e-mācības, tiešsaistes spēles un straumēšana. Datu pārsūtīšanas joslu platumi kļuva piesātinātāki un radītais lietotāju reakcijas datu apjoms pieauga eksponenciāli (Kastouni & Ait Lahcen, 2020).

Atbilstoši pētnieku (Wang et al., 2020) uzskatiem zināšanu izguves sistēmu pakalpojumu mūsdienīga arhitektūra sastāv no trim galvenajiem slāņiem (skat., 2.1.att.). Datu ieguves un uzglabāšanas slānī datu avoti tiek apkopoti, glabāti un apstrādāti izklaidētā datņu sistēmā vai datu bāzu sistēmā. Datu apstrādes slānī tiek izmantoti dažādi apstrādes risinājumi atbilstoši datu formām. Datu analīze galvenokārt balstās uz liela mēroga mašīnmācīšanās tehnoloģijām, kas spēj izgūt datu potenciālo vērtību. Visbeidzot tiek izmantoti vizualizācijas rīki, lai uzrādītu rezultātus datu pakalpojumu patērētājiem. Lietojumprogrammu slānī tiek iekļauti datu tehnoloģiju lietojumi dažādās sfērās. Zināšanas izguvju sistēmu datu mākoņskaitļošanas pakalpojumos apstrādei tiek izmantota programmatūra un infrastruktūra, kas veidota uz mākoņa modeļa darbības pamatprincipiem

(angliski: SaaS - Software as a Service, PaaS - Platform as a Service, IaaS - Infrastructure as a Service).



2.1.attēls. Lielo datu servisa arhitektūra (adaptēts no Wang *et al.*, 2020).

Šādu ietvara arhitektūru uzrāda pētnieki (Wang *et al.*, 2020), to veido vairāki slāņi, katrā iekļaujot noteiktas tehnoloģijas, kas spēj nodrošināt atbilstošā slāņa funkcionālās prasības. Ietvara arhitektūra (skat., 2.1.att.) iekļauj tehnoloģiskos apsvērumus kā datu glabāšanas, mākoņskaitļošanas pakalpojumus un datu apstrādi, kurus ir nozīmīgi aplūkot sīkāk.

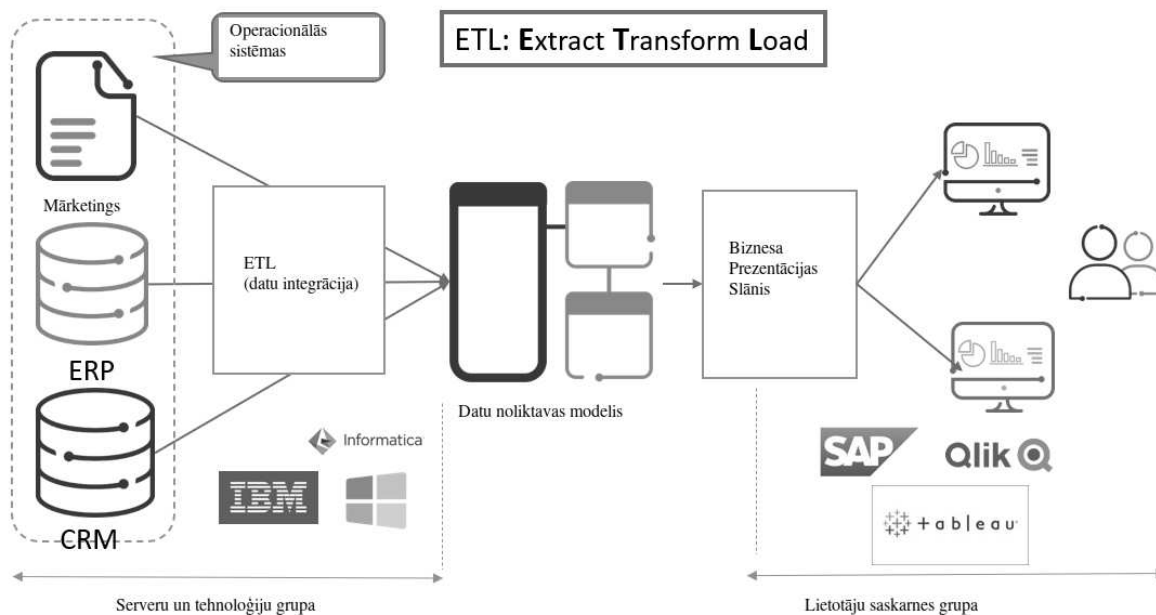
2.1. TEHNOLOĢISKIE APSVĒRUMI

2.1.1. DATU GLABĀŠANA

Datu glabāšana zināšanu izguves sistēmās galvenokārt ietver pakešu datus un dinamiski straumētus datus. Pakešu dati tiek glabāti statiskā formā, savukārt straumēšanas dati ir nepārtrauktu reāla laika datu ierakstu secība. Daļa no straumēšanas datiem bieži netiek glabāti, elementi tiek dzēsti pēc to apstrādes (Karunaratne *et al.*, 2017).

Straumētu datu pārraides nestabilitātes ietekmē, straumēšanas datu uzkrāšana atšķiras no tradicionālās pakešu datu uzkrāšanas. Pakešu datiem no vairākiem datu avotiem bieži tiek izmantoti ETL (angliski: Extract-Transform-Load) rīki, lai realizētu dažādu datu tipu pārsūtīšanu un apkopošanu. ETL procesā tiek iegūti dati no avotiem, pēc tam pārveidoti un ielādēti datu glabātuvē (skat., 2.2.att.). Šajā procesā tiek izņemti bojāti vai

neatbilstoši dati, izmantojot datu apstrādes procesus, piemēram, savienošanu, pārveidošanu un tīrīšanu. Plaši izmantotie ETL rīki ir Kettle, Datastage, Informatica (Wang et al., 2020).



2.2.attēls. ETL process (adaptēts no Data Valley, 2021).

Zināšanu izguves sistēmās straumētu datu uzkrāšanai, ar prasību darbībai būt pietuvināti reālam laikam, ir nepieciešami risinājumi, kas spētu nodrošināt tūlītēju darbību, kļūdu toleranci, stabilitāti un uzticamību. Viens no risinājumiem ir Flume (Apache Flume, 2022), pret defektiem noturīga straumēšanas sistēma, kas spēj apkopot un pārsūtīt lielu apjomu datus no vairākiem avotiem uz centralizētu glabātuvi. Flume parasti tiek izmantota kopā ar Hadoop (Apache Hadoop, 2021) ekosistēmu un darbojas kā programmatūra starp datu avotiem un datu patērētājiem (Shu et al., 2017). Alternatīvs risinājums ir Kafka (Apache Kafka, 2022), universāla atvērta pirmkoda ziņojumapmaiņas sistēma, ko galvenokārt izmanto, lai izveidotu reāllaika datu plūsmas un straumēšanas lietojumprogrammas.

Pēdējās desmitgadēs informāciju tehnoloģiju risinājumiem bieži tika izmantotas relāciju datu bāzes un strukturētas datu pārvaldības metodes (Deagustini et al., 2013). Savukārt lielais datu apjoms izvirzījis jaunas prasības datu glabāšanai un sistēmām biežāk tiek izmantotas dalītās (angliski: distributed) datņu sistēmas: NoSQL, NewSQL un citas datu pārvaldības sistēmas. HDFS (angliski: Hadoop Distributed File System) (Apache Hadoop, 2021) tiek uzskatīta par plašāk izmantoto lielu datu glabāšanas datņu sistēmu, kura atbalsta dublēšanu un mērogojamību paralēli dalītās arhitektūras sistēmās (Bok et al., 2017). Wang et al. (2020), apskatot biežāk izmantotās pieejas liela apjoma datu sistēmu

arhitektūras risinājumos, piedāvā uzskatāmu tabulu, kas atspoguļo NoSQL datubāzes, kas tiek pielietotas sistēmās datu uzglabāšanai (skat. 2.3.att.).

NoSQL datubāzes	Tips	Iezīmes	Priekšrocības	Trūkumi	Lietošanas scenāriji
Redis	Atslēgu-Vērtību	Atslēgu-vērtību pāri; Vienkārša datu struktūra;	Labā rakstīšanas-lasīšanas veiktspēja; Datu noturība	Bez automātiskas kļūdu noturības un datu atgūves atbalsta; Neatbalsta tiešsaistes paplašināšanu	Lietotāju datu glabāšanai, kas saistīti ar ID (atslēgu), piemēram, sesijas, konfigurācijas datus, parametri
Memcached		Augsta mērogojamība; Zema vaicājumu veiktspēja.	Atmiņas bāzēta datu apstrāde; Daļēti un mērogojama; Augsts apstrādes ātrums	Maza viena kešatmiņas ietilpība; Neatbalsta datu noturību	
MongoDB	Dokumentu Orientētas	Dokumentu bāzēta datu apstrāde;	Augsts piekļuves ātrums; Vairāku datu tipu atbalsts; Augsta vaicājumu efektivitāte	Zema lasīšanas-rakstīšanas efektivitāte; Aizņem daudz vietas failu sistēmā	Piemēroti žurnālu datņu glabāšanai un apstrādei tuvināti reālam laikam
CouchDB		Vairāku formātu atbalsts - JSON, XML.	Pieejamība un nepārtrauktība; Laba pielāgojamība	Zems vaicājumu izpildes ātrums; Augsta uzturēšanas sarežģītība	
Hbase	Kolonnas Orientētas	Kolonnas bāzēti glabāšanas modeļi; Augsta savienojamība vienā kolonnas klasē	Augsti ielādes ātrumi; Piemēroti lielu datu glabāšanai; Efektīva saspiešana	Zems saliktu vaicājumu izpildes ātrums; Tiešā veidā neatbalsta SQL vaicājumus	Piemērotas augstas vienlaicīgu darbību sistēmās ar lieliem datiem; Tuvināti reālā laika lasīšanas/rakstīšanas darbībām
Cassandra			Elastīga mērogojamība; Pielāgojama datu shēma; Vairāku sarakstu struktūra	Neatbalsta ACID transakcijas un atomiskas darbības	
Neo4j	Grafu bāzētas	Grafu bāzēti datu glabāšanas formāti; Augsta datu savienojamība starp ierakstiem;	Augsts vaicājumu izpildes ātrums; Augstas veiktspējas grafiskais algoritms	Neatbalsta lielus grafu nodalījumus	Piemērotas relāciju datu glabāšanai, rekomendāciju sistēmām, kopienu tīmekļa vietnēm
GraphDB		Efektīva vaicājumu apstrāde	ACID transakcijas un atomiskas darbības; Vairāku modeļu objekti	Augsta sarežģītība	

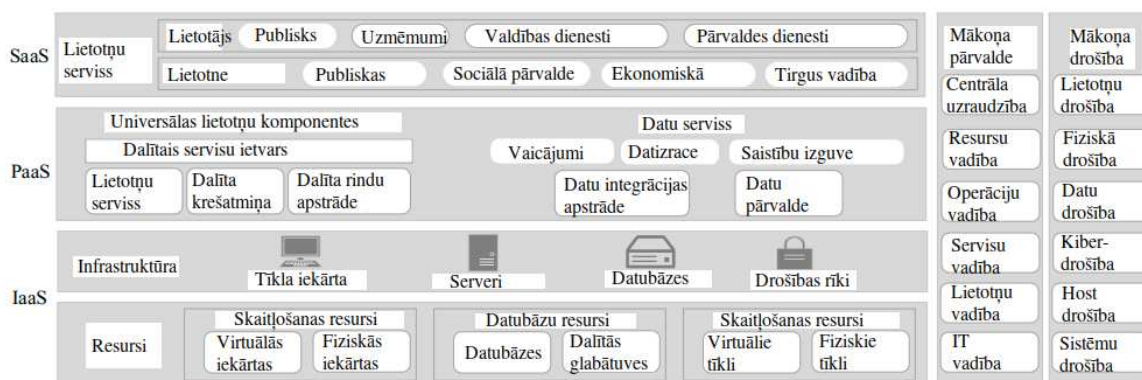
2.3.attēls. NoSQL datubāzu tipi (adaptēts no Wang et al., 2020).

NoSQL apzīmē datubāzu kopumu, kuras datu pārvaldībai netiek izmantotas relāciju saites. NoSQL datu modelis visbiežāk ietver atslēgu un vērtību pārus, kolonnas saimes vai dokumentus. Lietojot atslēgu-vērtību pāru modeļu struktūru, datu kolīzijas ir retākas un datu modeļi ir vieglāk ieviest. Uz dokumentiem balstīts datu modelis, izmanto unikālu atslēgu, lai identificētu katru dokumentu (Rich, 2017). NoSQL datubāzes ir paredzētas operatīvām prasībām — reāllaika lietojumprogrammām. NoSQL nodrošina augstas veiktspējas, veiklu informācijas apstrādi lielos apjomos. Dati tiek glabāti nestrukturēti vairākos apstrādes mezglos, kā arī vairākos serveros, tādēļ NoSQL izplatītā datu bāzes infrastruktūra ir viens no piemērotākajiem risinājumiem zināšanu izguves sistēmām ar lielām datu noliktavām. Lai apmierinātu pieprasījumu pēc datu pārvaldības un risinātu datu pieaugošo savstarpējo atkarību un sarežģītību, izveidotas NoSQL datubāzes, kuras nodrošina pilnvērtīgāku datu kopu pārvaldību un analīzi (Rich, 2017).

Zināšanu izguves sistēmu datu glabāšanu iespējams izveidot pielietojot konkrētam lietošanas gadījumam piemērotākās tehnoloģijas (skat., 2.3.att.), savukārt apstrādei nepieciešamās skaitļošanas resursu jaudas var nodrošināt mākoņskaitļošanas pakalpojumi.

2.1.2. MĀKOŅSKAITĻOŠANAS SISTĒMAS

Mākoņskaitļošanas tehnoloģiju priekšrocības ir tās dalītās apstrādes iespējas, izkliedētās datubāzes un virtualizācijas tehnoloģijas. Uz apjomīgiem datiem balstītas zināšanu izguves sistēmas izstrādē ir nepieciešams izmantot mākoņskaitļošanas platformas (Wang et al., 2020). 2.4. attēlā uzrādīti pieejamie mākoņskaitļošanas arhitektūru veidi - SaaS, PaaS, IaaS. SaaS - lietotājiem nav nepieciešams uzturēt savu programmatūru vai izstrādāt specifiskus risinājumus, mākoņpakalpojumi pieejami noteiktu uzdevumu izpildei un pēc uzdevuma veikšanas nepieciešams veikt apmaksu atkarībā no patēriņa. PaaS - nodrošina mērogojamu, sadalītu un kļūdu noturīgu (angliski: fault tolerant) mākoņpakalpojumu programmēšanas platformas datu apstrādei. IaaS - mākoņskaitļošanas pakalpojumu sniedzēji nodrošina konfigurējamus skaitļošanas resursus (procesorus, datu glabātuves, tīklus un lietojumprogrammas u.c.) (IBM Inc, 2021a).



2.4. attēls. Mākoņskaitļošanas pakalpojumu arhitektūra (adaptēts no Wang et al., 2020).

Zināšanu izguves sistēmu izstrādē mākoņskaitļošanas platformas tiek izmantotas, lai izveidotu konfigurējamu skaitļošanas resursu lietojuma modeli, kuru iespējams viegli un ātri konfigurēt un pārvaldīt (Armbrust et al., 2010). Skaitļošanas arhitektūras komponentes nepieciešams rūpīgi plānot un izstrādāt, ņemot vērā sistēmas īpašības, izmaksas, ātrdarbību un nepieciešamo mērogojamību.

Zināmākie un lielākie mākoņpakalpojumu sniedzēji ir Amazon Web Services, Microsoft Azure, Google Cloud Platform. Pakalpojumu sniedzēji piedāvā infrastruktūru, atbilstoši dažādām datu izstrādes vajadzībām, kas spēj nodrošināt pilnu zināšanu izguves sistēmas izveides un datu apstrādes ciklu. Mašīnmācīšanās algoritmu izstrādei pakalpojumu sniedzēji atbalsta regresijas, klasifikācijas, klasterizācijas un citus algoritmus, kā arī piedāvā zināmas mācīšanās sistēmas, piemēram, TensorFlow, MXNet, Caffe, PyTorch (Y. Jia, 2021.; MXNet, 2021; PyTorch, 2021; TensorFlow, 2022).

Zināšanu izguves ietvara izstrādei nozīmīgs nosacījums ir izmantot atbilstošu mākoņskaitļošanas pakalpojumu sniedzēju. Skaitļošanas jaudas ir nepieciešamas ne tikai tehnoloģisko procesu nodrošināšanai, bet arī datu apstrādei. Izveidot, uzturēt un nodrošināt atbilstošus risinājumus tikai ar izstrādātāja vai pasūtītāja resursiem ir potenciāli iespējams, tomēr šāda veida pieeja palielinātu tehnisko procesu sarežģītību un apmēru.

2.1.3. LIETOTĀJU REAKCIJU DATU PRIEKŠAPSTRĀDE

Datu priekšapstrādes procesā neapstrādātos datus ir nepieciešams attīrīt un izņemt neizmantotās un/vai nelietderīgās datu kopas. Zināšanu izguves sistēmas ar lielu datu apjomu, apstrādes un analīzes izmaksas, izmantojot visus pieejamos un uzkrātos datus, ir augstas (Xindong Wu et al., 2014).

Pakešu dati visbiežāk ir statistiski un datu apjoms ir liels, datu apstrāde tiek veikta, izmantojot apstrādes metodi, kas spēj veikt paralēlo skaitļošanu. 2004. gadā Google izstrādāja MapReduce (Wang et al., 2020), kas ir programmēšanas ietvars lielu datu kopu apstrādei. MapReduce ļauj lietotājiem izveidot sarežģītus aprēķinus lielām datu kopām nodrošinot sinhronizāciju, kļūdu toleranci, uzticamību un pieejamību. MapReduce parasti sadala ievades datu kopu neatkarīgos blokos, kurus apstrādā paralēli uzdevumi (IBM Inc, 2021b).

Straumētu datu apstrādes pieeja ir piemērota tādu datu apstrādei, kam nepieciešama gandrīz tūlītēja atbilde, tādēļ strauvētu datu apstrādei nepieciešamas tādas datu apstrādes sistēmas, kas spēj sasniegt lielu izpildes ātrumu. Pieejami vairāki strauvētu datu apstrādes produkti, piemēram, Storm ir atvērtā koda reāllaika strauvētu datu apstrādes sistēma, kura spēj nodrošināt augstu veiktspēju un zemu izpildes aizkavi un ir piemērota strauvētu datu apstrādei. Storm alternatīva ir Apache Samza strauvētu datu apstrādes sistēma, kas spēj efektīvi apstrādāt apjomīgu lietotāju reakcijas datu daudzumu. Samza un Kafka kombinācija nodrošina iespēju labāk izmantot abu sistēmu priekšrocības - Kafka var

nodrošināt kļūdu toleranci, datu buferizāciju un stāvokļa glabāšanu (*Apache Kafka*, 2022; *Apache Samza*, 2022; *Apache Storm*, 2021; Oussous et al., 2018).

Atsevišķi datu priekšapstrādes uzdevumi var ietvert gan pakešu datu apstrādi, gan straumētu datu apstrādi, daļa datu apstrādes sistēmu atbalsta gan pakešu, gan straumētu datu apstrādes veidus. Apache Spark ir pakešu datu apstrādes sistēma ar straumētu datu apstrādes iespējām. Spark Streaming komponente piedāvā metodi - Micro-batch, kas nepārtrauktas plūsmas apstrādā kā mikropakešu datu sēriju un apstrādā mikropakešu uzdevumus (*Apache Spark*, 2021). Storm un Spark platformu salīdzinājums redzams 2.5 attēlā. No Oussous et al. (Oussous et al., 2018) piedāvātā salīdzinājuma redzams, ka Spark ir lielāka datu aizkave kā Storm platformai, toties tā spēj nodrošināt gan straumētu datu apstrādi, gan pakešu datu apstrādi tuvināti reālam laikam. Kā alternatīvu Spark platformai var minēt arī Flink, platformu, kura tāpat kā Spark spēj nodrošināt gan straumētu, gan pakešu datu apstrādi tuvināti reālam laikam (*Apache Flink*, 2021).

Pazīme	Projekts	
	Sprak	Storm
Organizācija	UC Berkeley	BackType, Twitter
Veids	Atvērtā koda	Atvērtā koda
Programmēšanas valoda	Scala	Coljure
Atbalstītās valodas	Java, Python, R, Scala	Visas
Izpildes modelis	Pakotņu, straumes	Straumētu datu
Aizkave	Dažu sekunžu aizkave (atkarībā no pakotnes izmēra)	Dažu sekunžu daļu aizkave
Pārvaldes veids	Raksta datus glabātuvē un nepieciešami stāvokļa aprēķini	Pielieto trident un bezstāvokļa skaitļošanu
Kļūdu noturība	Atbalsta "exactly once" pieeju	Atbalsta "exactly once", "at least once" un "at most once" veidus
Straumētu datu avoti	HDFS	Spout
Straumētu datu apstrāde	Windows Operācijas	Bolts
Straumētu datu datu veidi	Dstream	Tuple
Uzraudzība	Vienkārša uzraudzība ar ganglia	Apache Ambari
Resursu pārvaldes lietotnes	Mesos un Yam	Mesos
Hadoop	HDP, CDH, MapR	HDP

2.5.attēls. Spark un Storm salīdzinājums (adaptēts no Oussous et al., 2018).

Zināšanu izguves sistēmu datu apstrādei lielos apjomos nepieciešami skaidri definēti biznesa mērķi, kuri ietver ne tikai ātru apstrādi, bet arī radītu pievienoto vērtību. Pievienoto vērtību no datiem iegūst tos analizējot un vizualizējot, kā arī izmantojot mašīnmācīšanās algoritmus, lai no datiem izgūtu lietderīgas zināšanas, kas atbalstītu nākotnes tendenču prognozēšanu un situāciju analīzi. Datu pakalpojumu patērētājiem pēc apstrādes un analīzes ir jāiegūst lietderīgas zināšanas (Oussous et al., 2018).

Zināšanu izguves sistēmās datu vizualizācijai bieži tiek izmantotas tabulas un attēli, lai gala patērētājiem sniegtu informāciju par datiem. Izmantojot vizuālos efektus, datus var attēlot saprotamā un intuitīvā formā, tādējādi palielinot izpratni informācijas patērētāju starpā. Izgūtās datu vizualizācijas, lielo datu tendences, datu modeļus, korelāciju analīzes un citu informāciju datu analītiķi izmanto, lai datus pētītu padziļināti no vairākām dimensijām un uzlabotu datu analīzi (Oussous et al., 2018).

Datu glabāšana, apstrāde, vizualizācija un mākoņskaitļošanas resursu pielietošana ir tikai daļa no zināšanu izguves sistēmu tehnoloģiskajiem apsvērumiem. Papildus uzskaitītājam procesā iesaistītas arī lietojumprogrammas vai lietu interneta iekārtas, kuras veic datu uzkrāšanu. Tehnoloģiskie kritēriji lietotnēm būtiski atšķirsies no lietojumprogrammas biznesa mērķiem un pielietojuma. Zināšanu izguves ietvariem jāatbilst arī datu uzkrāšanas un glabāšanas regulējošajiem tiesiskajiem standartiem, kā arī tiem jāiekļauj preventīvi un korektīvi risinājumi kiberaudraudējumu scenārijiem, turpmāk aprakstītiem kā ietekmējošie kritēriji.

2.2. ZINĀŠANU IZGUVE UN IETEKMĒJOŠIE KRITĒRIJI

Zināšanu izguves sistēmu viens no mērķiem ir uzkrāt datus, lai tos apstrādātu, analizētu, radītu pievienoto vērtību un pieņemtu lēmumus. Savukārt personu datu aizsardzības regula nosaka, ka noteiktiem un definētiem mērķiem jāizmanto tikai minimāli nepieciešamās datu kopas (EP, 2016). Personu datu aizsardzības regulas principi attiecas uz personas datu apstrādi, organizācijām ir jāizvērtē vai konkrētas personas datu izmantošana atbilst attiecīgā datu subjekta saprātīgām sagaidāmām prasībām, kas savukārt ir pretrunā ar liela apjoma datu (tai skaitā zināšanu izguves sistēmu) uzkrāšanas sistēmu praksi. Datu subjektam ir nepieciešams izskaidrot datu uzkrāšanas mērķus, izmantojot skaidru un saprotamu privātuma politiku, kas ir īsa, uzrakstīta vienkāršā valodā un viegli pieejama (Jeren, 2020).

2.2.1. VISPĀRĪGĀ PERSONU DATU AIZSARDZĪBAS REGULA

Vispārīgā datu aizsardzības regula (VDAR) ir regula ES tiesību aktos, kura nosaka aizsardzību un privātumu Eiropas Savienībā (ES) un Eiropas Ekonomikas zonā (EEZ). VDAR ir svarīga ES privātuma tiesību un cilvēktiesību sastāvdaļa. Regula attiecas arī uz personas datu pārsūtīšanu ārpus ES un EEZ zonām. VDAR galvenais nolūks ir uzlabot personu kontroli un tiesības pār saviem personas datiem un vienkāršot starptautiskās uzņēmējdarbības normatīvo vidi (EP, 2016).

Zināšanu izguves sistēmu datu uzkrāšana, glabāšana, analīze un izmantošana, lai no tiem izgūtu lietderīgas zināšanas, ir pretrunā tam, ko personām garantē Eiropas Savienības Pamattiesību harta un VDAR. Atbilstoši tām personas dati ir jāaizsargā un jāapstrādā godīgā veidā ar definētiem un konkrētiem mērķiem, un tos nedrīkst glabāt ilgāk nekā nepieciešams. Uzņēmumiem iekšējos procesos un pakalpojumos jāiekļauj drošības, privātuma un tehniskie pasākumi, lai garantētu datu subjektiem viņu tiesības atbilstoši VDAR nosacījumiem (EP, 2016).

Arī tad, ja organizācijas un uzņēmumi pieņem, ka tie atbilst VDAR nosacījumiem, jo izmanto personas datus likumīgi un godīgi, var rasties sarežģījumi, ja datus regulāri nedzēš. Saskaņā ar 5. panta 1. punkta e) apakšpunktu personas datus nedrīkst uzglabāt ilgāk nekā nepieciešams (EP, 2016). Organizācijām būtu vēlams noteikt datu glabāšanas periodus un ieviest automātisku datu dzēšana. Datu subjektiem ir tiesības piekļūt personas datiem, tos labot un dzēst, kā arī ierobežot to apstrādi, ko nosaka VDAR 17. pants. Tātad zināšanu izguves sistēmām organizācijās jāspēj apstrādāt lielais datu apjoms, kas tiek glabāts sistēmās, lai atrastu vai dzēstu subjektam piederošos datus.

Viens no variantiem, kā zināšanu izguves sistēmas padarīt atbilstošas VDAR, ir datus anonimizējot vai pseidonimizējot. VDAR uzsver atšķirību starp pseidonimizāciju un anonimizāciju. Saskaņā ar definīciju, kas atrodama VDAR 4.panta 5. apakšpunktā: *“pseidonimizācija ir personas datu apstrāde, ko veic tādā veidā, lai personas datus vairs nav iespējams saistīt ar konkrētu datu subjektu bez papildu informācijas izmantošanas, ar noteikumu, ka šāda papildu informācija tiek turēta atsevišķi un tai piemēro tehniskus un organizatoriskus pasākumus, lai nodrošinātu, ka personas dati netiek saistīti ar identificētu vai identificējamu fizisku personu”* (EP, 2016). Anonimizācija attiecas uz praksi padarīt datus neidentificējamus tādā veidā, ka nav iespējams iegūt datu subjekta identitāti. Ja dati tiek anonimizēti, VDAR regulējums uz tiem neattiecas.

Saskaņā ar VDAR 22. panta 1. punktu personām ir tiesības netikt pakļautām automatizētai lēmumu pieņemšanai, tostarp profilēšanai. Profilēšana visbiežāk tiek veikta izmantojot mašīnmācīšanās algoritmus, tādējādi VDAR ierobežo arī zināšanu izguves sistēmu mašīnmācīšanās algoritmus. Organizācijas joprojām var izmantot automatizētu lēmumu pieņemšanu, ja ir nepieciešams izpildīt vai noslēgt līgumus, ko atļauj dalībvalsts tiesību akti vai, ja datu subjekts ir devis nepārprotamu piekrišanu. Zināšanu izguves sistēmās profilēšana bieži ir personām neredzama, un tā ir prakse, pret kuru datu aizsardzības likums cenšas aizsargāt personas, liekot organizācijām nodrošināt skaidrību un iespēju atteikties no automatizētas profilēšanas (EP, 2016).

Nozīmīgs ietekmējošais faktors ir iespēja datu subjektam saņemt saprotamu skaidrojumu. Datu subjektam būtu jādod iespēju izprast datu apstrādes loģiku un tās ietekmi. VDAR 13. panta 2. punkta f) apakšpunkts nosaka datu subjektiem tiesības saņemt jēgpilnu informāciju par iesaistīto loģiku, kā arī šādas apstrādes nozīmi un paredzamajām sekām (EP, 2016). Tātad, ja zināšanu izguves sistēmas algoritmos tiek izmantoti personu dati, tad datu subjektam ir tiesības saņemt saprotamu šo algoritmu izpildes izskaidrojumu - algoritmus jāizskaidro subjektiem bez zināšanām par to darbību.

Dati ir nozīmīgi visa veida organizācijām, lai analizētu datu aktīvus un uzlabotu procesus un piedāvātos produktus. Taču kontroles sniegšana datu subjektam, kā to paredz VDAR, nozīmē, ka organizācijām ir jāņem vērā personu datu regulas nosacījumi uzkrājot un analizējot datus. Vispārīgā personu datu aizsardzības regula būtiski ietekmē zināšanu izguves sistēmas un to funkcionalitāti, ja vien dati netiek pilnībā anonimizēti.

2.3.2. KIBERDROŠĪBA

Attīstoties internetam, kiberuzbrukumi mainās un kiberdrošības situācija nav optimistiska (Xin et al., 2018). Datoru drošība, kiberdrošība vai informācijas tehnoloģiju drošība ir datorsistēmu un tīklu aizsardzība pret informācijas izpaušanu, to aparatūras, programmatūras vai elektronisko datu zādībām vai bojājumiem, kā arī no pakalpojumu pārtraukšanas vai nepareizas novirzīšanas (Schatz et al., 2017).

Kiberdrošības joma kļūst arvien nozīmīgāka, jo nepārtraukti pieaug atkarība no datorsistēmām, interneta un bezvadu tīkla standartiem, piemēram, Bluetooth un Wi-Fi, kā arī "viedo" ierīču, tostarp viedtālrunu, televizoru un dažādas ierīces, kas veido "lietu internetu". Kiberdrošība ir arī viens no nozīmīgākajiem izaicinājumiem mūsdienu pasaulē

tās sarežģītības dēļ gan politiskā lietojuma, gan tehnoloģiju ziņā (Xin et al., 2018; Schatz et al., 2017).

Jebkurš zināšanu izguves sistēmas projekts ietver ievērojamu datu apjomu, kas var iekļaut arī sensitīvu vai personu identificējošu saturu. Katrai organizācijai, kas glabā datus, ir jānodrošina datu konfidencialitāte, integritāte un pieejamība. Datu drošībai vēlams ņemt vērā vienkāršus principus (skat., 2.6.att.) - datu drošību, pieejas kontroli un datu aizsardzību (Bigelow, 2021).



2.6.attēls. Datu drošība (adaptēts no Bigelow, 2021).

Eksistē vairākas labās prakses datu drošībai, piemēram, vismazākās privilēģijas principi, datu šifrēšana, konstanta monitorēšana u.c.. Nozīmīgi ir izstrādāt zināšanu izguves sistēmas arhitektūru, iekļaujot drošības prasības jau izstrādes procesā. Būtiski ir arī pilnībā saprast datu plūsmas zināšanu izguves sistēmas infrastruktūrā un samazināt uzbrukuma virsmas izmērus (angliski: attack surface). Izstrādājot sistēmas prasības, ieteicams izmantot kādu drošības prasību noteikšanas metodoloģiju, piemēram, SQUARE (Mead, 2006), MOSRE (Salini & Kanmani, 2013) u.c.

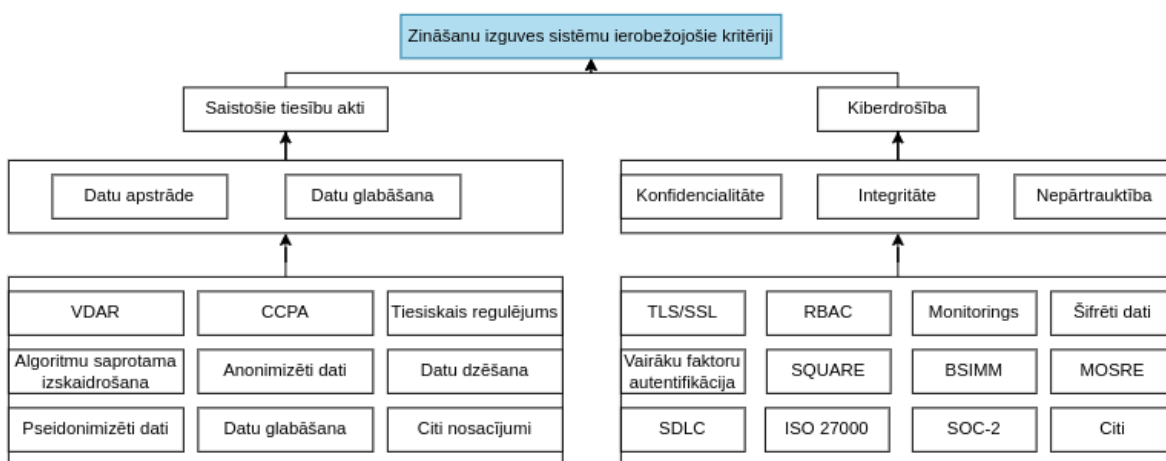
Obligāta prasība ir autentifikācijas un autorizācijas nodrošināšana gadījumos, kad tiek veikta piekļuve zināšanu izguves sistēmas infrastruktūrai vai tās datiem. Ja dati ir uzskatāmi par īpaši nozīmīgiem, piemērots risinājums ir divu faktoru autorizācija. Sistēmu un dalībnieku pieejas tiesību definēšanā nepieciešams izmantot vismazāko privilēģiju pieeju, piemēram, ja zināšanu izguves sistēmas noteiktai komponentei ir nepieciešams datus nosūtīt uzglabāšanai (sensori, novērošanas kameras u.c.), tad šo objektu pieejas tiesības nedrīkst saturēt lasīšanas, dzēšanas un labošanas privilēģijas (Xin et al., 2018; Schatz et al., 2017; Bigelow 2021).

Zināšanu izguves sistēmā uzkrātos lietošanas reakcijas datus vēlams glabāt šifrētā formātā gan tos glabājot, gan pārvietojot starp sistēmas komponentēm. Šādā scenārijā, ja

nesankcionētai personai izdodas piekļūt datu kopai, šifrētie dati nebūtu lietojami. Šifrējot datus pārvietošanas laikā obligāti jāizmanto tādas metodes kā Transport Layer Security (TLS) vai Secure Sockets Layer (SSL, novecojusi). Zināšanu izguves sistēmai nevajadzētu eksistēt komponentēm, kurām ir tiesības iegūt datus no glabātuves neizmantojot TLS/SSL (Xin et al., 2018; Schatz et al., 2017; Bigelow 2021).

Kā preventīvs drošības nosacījums ir zināšanu izguves sistēmas infrastruktūras monitoringa stratēģijas izveide, kas konstanti uzrauga datu glabātuves un skaitļošanas infrastruktūru. Mākoņskaitļošanas resursu uzraudzības rīkus piedāvā vairāki pakalpojumu sniedzēji, piemēram, Amazon CloudWatch, Azure Monitor un Google Cloud Operations. Rīki spēj apkopot monitoringa un operacionālos datus, un izveidot pārraudzības metrikas (Xin et al., 2018; Schatz et al., 2017; Bigelow 2021).

Zināšanu izguves sistēmu izstrādi ietekmē gan noteiktā darbības reģionā spēkā esošais tiesiskais regulējums, gan kiberdrošības faktori (piemēram, VDAR Eiropā, CCPA ASV) (skat., 2.7.att.). Izstrādājot zināšanu izguves sistēmu drošības prasības nepieciešams iekļaut jau sistēmas plānošanas laikā, kā arī nepieciešams izmantot kādu drošības prasību noteikšanas metodoloģiju, var būt nepieciešams arī iegūt atbilstošus sertifikātus (ISO 27000 Eiropā, vai SOC-2 ASV).



2.7.attēls. Zināšanu izguves sistēmu ietekmējošie kritēriji (autora veidots).

Datu drošības apsvērumi un kritēriji (skat., 2.7.att.) ir nozīmīgi ikvienai organizācijai, daļā uzskaitītais nav uzskatāms par pietiekamiem nosacījumiem zināšanu izguves sistēmu drošībai. Ikvienas zināšanu izguves sistēmas izstrādē jau sākotnējā arhitektūras izstrādes procesā nepieciešams iesaistīt kvalificētus kiberdrošības speciālistus un iekļaut drošības prasības sistēmas izstrādes dzīves ciklā.

2.4. NODAĻAS KOPSAVILKUMS UN SECINĀJUMI

Darba pirmajā daļā apskatītajos un salīdzinātajos zināšanu izguves ietvaros detalizēti netiek aprakstītas tehnoloģijas, datu apstrādes un glabāšanas iespējas, un zināšanu izguves sistēmas ierobežojošie kritēriji, kas rada nepieciešamību noteikt zināšanu izguves ietvaru tehnoloģiskos nosacījumus. 2.1. tabulā sniegts tehnoloģisko kritēriju un ietekmējošo faktoru apkopojums.

2.1. tabula. Zināšanu izguves ietvaru izstrādes tehnoloģiskie kritēriji (autora veidota)

Pazīme	Raksturojums
VDAR un tiesiskais regulējums.	Nepieciešams ņemt vērā, lai personas dati netiktu pielietoti zināšanu izguves sistēmās. Atkarīgs no ģeogrāfiskā novietojuma un spēkā esošā tiesiskā regulējuma, ES regulē VDAR.
Kiberdrošība	SQUARE, MOSRE, BSIMM, SSL/TLS, draudu modelēšana, resursu uzraudzības rīki.
Datu glabāšana	NoSQL, NewSQL, SQL, Flume, Hadoop, HDFS, tehnoloģiju kopums atkarībā no lietošanas gadījuma.
Datu apstrāde	Samza, Kafka, Spark, MapReduce, Storm, Flink, trešo pušu servisi.
Skaitļošanas tehnoloģijas	Konteinerizācijas tehnoloģijas, SaaS, PaaS, IaaS un mākoņskaitļošanas pakalpojumi.
Arhitektūra	Organizēta slāņos: lietotņu, apstrādes, uzglabāšanas, drošības, skaitļošanas, lietojumprogrammu saskarņu.

Liela apjoma datu apstrādes tehnoloģijas tiek pielietotas zināšanu izguvei, datu analīzei un datu koplietošanai un radījušas vērā ņemamus ekonomiskos ieguvumus. Iegūtās zināšanas var nodrošināt lēmumu pieņemšanas stratēģijas sociālajai un ekonomiskajai attīstībai. Dati un mākoņskaitļošana tiek uzskatīti par galvenajiem faktoriem, kas ietekmē nākotnes ekosistēmu veidošanos.

Atbilstoši Wang et al. (Wang et al., 2020) zināšanu izguves un lielo datu pakalpojumu mūsdienīga arhitektūra, sastāv no trim galvenajiem slāņiem: datu ieguves, apstrādes un vizualizācijas.

Datu glabāšana, apstrāde, vizualizācija un mākoņskaitļošanas resursu pielietošana ir tikai daļa no zināšanu izguves sistēmu tehnoloģiskajiem apsvērumiem, procesā iesaistītas

arī lietojumprogrammas un/vai lietu interneta iekārtas. Tehnoloģiskie apsvērumi lietotnēm būtiski atšķirsies no lietojumprogrammas biznesa mērķiem. Zināšanu izguves sistēmām jāatbilst arī datu uzkrāšanas un glabāšanas regulējošajiem tiesiskajiem standartiem, kā arī tām jāiekļauj preventīvi un korektīvi risinājumi kiberapdraudējumu scenārijiem.

Personu datu aizsardzības regula nosaka, ka noteiktiem un definētiem mērķiem jāizmanto tikai minimāli nepieciešamās datu kopas. Tas nozīmē, ka organizācijām ir jāizvērtē, vai konkrētas personas datu izmantošana atbilst attiecīgā datu subjekta saprātīgām sagaidāmajām prasībām, kas ir pretrunā ar liela apjoma datu uzkrāšanas sistēmu praksi.

Zināšanu ieguves ietvara no viedo sistēmu lietotāju reakcijas datiem izstrādes viens no nosacījumiem ir nepieciešamība izmantot atbilstošu mākoņskaitļošanas pakalpojumu sniedzēju. Izveidot, uzturēt un nodrošināt atbilstošus risinājumus tikai ar izstrādātāja vai pasūtītāja resursiem ir potenciāli iespējams, tomēr šāda veida pieeja palielinātu tehnisko procesu apjomu.

Darba otrajā daļā:

- identificēti būtiskie zināšanu izguves ietvaru arhitektūras elementi;
- ir apkopoti zināšanu izguves ietvaru izstrādes tehnoloģiskie kritēriji, tos apkopojot un detalizējot;
- ir izveidots zināšanu izguves ietvaru ietekmējošo kritēriju apkopojums, kurš pielietots UIS-KDF ietvara izstrādē;
- ir apkopoti likumdošanas akti, kuri ietekmē zināšanu izguves sistēmu izstrādi.

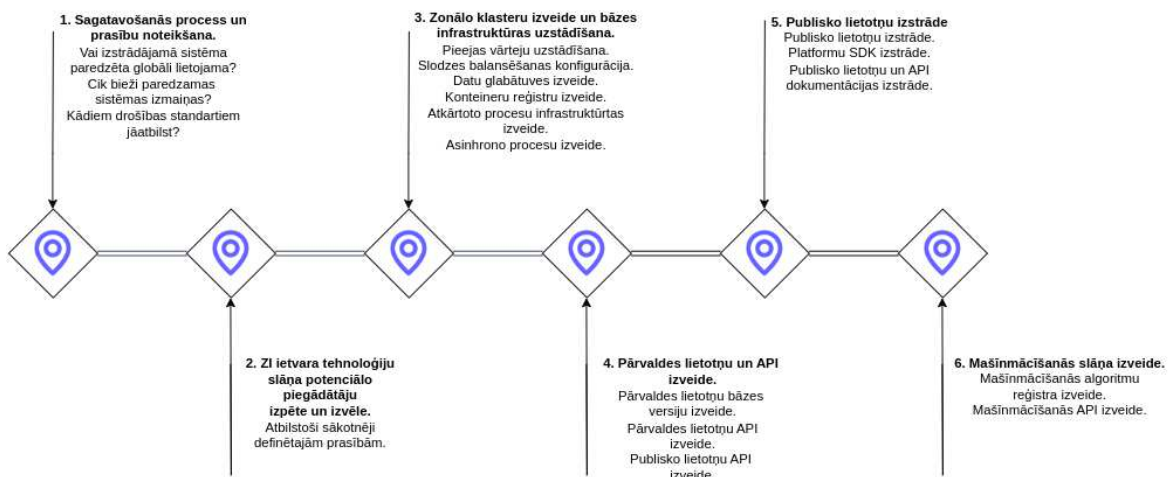
Zināšanu izguves sistēmas veido gan lielo datu servisi, gan tehnoloģijas, kuras nodrošina sistēmu darbību, kā arī tās ietekmē noteiktajā teritorijā spēkā esošie likumdošanas akti un kibernetikas prasības. Izstrādājot zināšanu izguves sistēmas no lietotāju reakcijas datiem jāņem vērā ietekmējošie faktori, kuri veido nepieciešamību pēc UIS-KDF ietvara, kas nodrošinātu izstrādes vadlīnijas. Darba otrās daļas rezultāti apkopoti zinātniskās publikācijās starptautiski citējamās datubāzēs iekļautajos izdevumos (Jansevskis & Osis, 2024b, 2024a).

3. UIS-KDF IETVARS

Zināšanu izguves sistēmas veido komplekss tehnoloģiju kopums, kas spēj nodrošināt sistēmu darbību. Izstrādājot zināšanu izguves ietvarus no lietotāju reakcijas datiem jāņem vērā darba otrajā daļā apskatīto faktoru kopums. Gan tehnoloģiju kopums, gan ietekmējošo faktoru apjoms padara zināšanu izguves sistēmu izstrādi par sarežģītu un laikietilpīgu procesu, kuru atvieglo zināšanu izguves ietvari, kas nodrošina izstrādes vadlīnijas. Konceptuāli zināšanu izguves ietvari no lietotāju reakcijas datiem tiek pielietoti zināšanu izguves sistēmu arhitektūras agrā izstrādes stadijā.

Zināšanu izguves sistēmas veido vairākas savstarpēji savienotas programmatūras sistēmas. Programmatūras sistēmas var būt vienas no sarežģītākajām konstrukcijām inženierzinātņu disciplīnās. Lielākā daļa programmatūras sistēmu daļēji īsteno to, kas jau ir uzbūvēts, un mēdz sekot zināmām vai gandrīz zināmām arhitektūrām. Lai gan lielākā daļa programmatūras sistēmu nav tādas kā, piemēram, Microsoft Windows, programmatūras izstrādes sarežģītība var strauji pieaugt. Tādējādi nozīmīga ir arhitektūras, modeļu un programmatūras ietvaru izmantošana. Modeļi nodrošina zināmus risinājumus atkārtotām problēmām, ar kurām saskaras izstrādātāji. Izmantojot modeļus, ietvaros var iekļaut atkārtoti lietojamas komponentes. Ietvari nodrošina izstrādātājiem instrumentus, lai efektīvāk izveidotu elastīgākas un mazāk kļūdām pakļautas lietojumprogrammas. Ietvari palīdz paātrināt izstrādes procesu, nodrošinot nepieciešamo funkcionalitāti. Veiksmīgas programmatūras izstrādes viens no nosacījumiem ir spēja nodrošināt ietvarus atkārtotai izmantošanai un problēmu nošķiršanai (Edwin, 2014).

Zināšanu izguve no lietotāju reakcijas datiem saistīta arī ar mašīnmācīšanās tehnoloģiju potenciālu un to nodrošinātajām iespējām (vairāk skat., XII pielikums). Uzņēmumu un organizāciju ekonomisko darbību, zināšanu izguves efektivitāti un zināšanu sabiedrības prasību pēc nepārtrauktām inovācijām risina informāciju tehnoloģijas un viedās programmatūras sistēmas. Uzņēmumiem un organizācijām biznesa lēmumu pieņemšanas procesā ir nepieciešama datos balstīta lēmumu pieņemšana.



3.1.attēls. Zināšanu izguves sistēmas izstrādes posmi (autora veidots attēls).

Ietvari palīdz paātrināt izstrādes procesu, veicina atkārtotu izmantošanu un problēmu nošķiršanu, un nodrošina izstrādātājiem instrumentus elastīgāku un mazāk kļūdām pakļautu lietotņu izstrādei. Darba autors zināšanu izguves sistēmu izstrādei definē vairākus secīgus posmus (skat., 3.1.att.) - prasību noteikšana, tehnoloģiju izvēle, infrastruktūras izveide, pārvaldes lietotņu un API izveide, publisko lietotņu izstrāde un mašīnmācīšanās pielietošana. Katram no posmiem raksturīgas noteiktas prasības, kuras nepieciešams pildīt, izstrādājot zināšanu izguves sistēmu.

3.1. ZINĀŠANU IZGUVES SISTĒMU IZSTRĀDES POSMI

3.1.1. PRASĪBU NOTEIKŠANA

Prasību noteikšana ir darbība, kura tiek veikta iteratīvi un lielākā detalizācijā risinājuma attīstības gaitā. Par augstākā līmeņa prasībām tiek uzskatīti biznesa mērķi un biznesa vīzija un ikvienam risinājumam ir jāpalīdz sasniegt organizācijas biznesa mērķus un jāatbilst tās vīzijai. Otrās prasību detalizācijas līmenis ir iesaistīto dalībnieku vajadzības un vēlmes un uz to pamatiem definētās biznesa prasības. Biznesa prasības nosaka nākotnes risinājuma konceptuālās arhitektūras un attīstības plāna definēšanu un ir pamats lēmuma pieņemšanai par projekta uzsākšanu (VARAM, 2017).

Uzsākot projektu, biznesa prasības tiek detalizētas līdz sistēmas prasībām, kas veido pamatu sistēmas attīstības uzdevumam. Ja sistēma sastāv no vairākām daļām, piemēram, infrastruktūras, infrastruktūras izmitināšanas pakalpojumiem un programmatūras, arī sistēmas prasības nepieciešams sadalīt atbilstošās daļās, piemēram, infrastruktūras prasībās, izmitināšanas pakalpojumu prasībās un programmatūras prasībās

(VARAM, 2017). Atsevišķām sistēmas daļām, prasību precizēšana un detalizēšana var turpināties arī sistēmas daļas izstrādes ietvaros.

Autora profesionālā pieredze liecina, ka prasību noteikšana ir nozīmīgs izstrādes posms, kurš raksturīgāks valsts iepirkumiem un pasūtījumiem, savukārt privātajā sfērā tas bieži netiek veikts detalizēti, jo uzņēmēji pielieto spējās izstrādes metodoloģijas un prasību definēšanu izstrādes procesa laikā. Savukārt jaunuzņēmumi, kuri vēl nav pilnībā validējuši savus pieņēmumus un noteikuši potenciālā produkta atbilstību tirgus pieprasījumam, prasību definēšanu bieži neveic vispār un izstrādes procesā izstrādā tikai minimālo dzīvotspējīgo produktu (MVP).

Nozīmīgākais prasību definēšanas izstrādes posma uzdevums ir noteikt kopējās organizācijas biznesa prasības, dalībnieku vēlmes un vajadzības kā arī sākotnēji identificēt nepieciešamās drošības prasības. Piemēram, zināšanu izguves sistēmām, kuras paredzēts pielietot globālā mērogā būs atšķirīgas infrastruktūras un drošības prasības salīdzinot ar tādām, kuras paredzētas lokālam tirgum.

3.1.2. TEHNOLOĢIJU IZVĒLE

Tehnoloģiju izvēle ir nozīmīgs izstrādes posms, tās nepieciešams izvēlēties atbilstoši prasību noteikšanas posmā identificētajām organizācijas vajadzībām. Izvēloties tehnoloģiju infrastruktūras pakalpojumus jānosaka veiktspējas un mērogojamības prasības, kuras būs nepieciešamas konkrētajai sistēmai. Vēlams identificēt sekojošos infrastruktūras parametrus (VARAM, 2017):

- skaitļošanas jaudas - izteiktas virtuālo procesora kodolu skaitā pie noteiktas takts frekvences;
- operatīvās un pastāvīgās atmiņas apjomi - izteikti baitos (piemēram, 64GB operatīvas atmiņas, 4 TB diska vietas);
- datu ievades un izvades operāciju skaits sekundē (piemēram, 60 000 IOPS);
- datu caurlaidība tīklam (piemēram, 1 Gbps). Būtiska var būt arī caurlaidība diskos vai datu glabāšanas iekārtām.

Minētie parametri ir tikai daļa no tehnoloģiju izvēli ietekmējošajiem faktoriem, ja zināšanu izguves sistēmai ir paredzams pastāvīgi pieaugošs lietotāju skaits vai noteiktā laika posmā pieaugošs lietotāju skaits, tad skaitļošanas jaudām ir jābūt arī automātiski palielināmām un samazināmām atbilstoši pieprasījumam. Kā arī, ja zināšanu izguves sistēmai paredzēts uzkrāt pasīvos lietotāju reakcijas datus, jāieplāno arī atbilstoši elastīga datu uzglabāšanas sistēma. Globāli lietojamu sistēmu infrastruktūru vēlams izvietot kādā

no lielākajiem infrastruktūras piegādātājiem, citādi nodrošināt skaitļošanas jaudu elastību var nebūt iespējams.

Neatbilstošu tehnoloģiju izvēle var novest pie nepietiekamas veikspējas produkta izstrādes un potenciāliem sarežģījumiem piegādes un uzturēšanas procesos. Piemēram, ja zināšanu izguves sistēmai paredzēts strādāt ar nestrukturētiem datiem, tad tradicionālo SQL datubāzu izvēle sagādās problēmas. Līdzīgi, ja zināšanu izguves sistēmas mērķa tirgus paredzēts, piemēram, ASV, tad tehnoloģiju infrastruktūras izvietošana Eiropā sagādās ātrdarbības problēmas.

3.1.3. INFRASTRUKTŪRAS IZVEIDE

Infrastruktūras izveide iekļauj tehnoloģiju izvietošana un konfigurāciju atbilstoši biznesa, veikspējas, mērogošanas, drošības un tehnoloģiju prasībām. Infrastruktūras izveide iekļauj serveru, zonālo klasteru, pieejas vārteju, slodzes balansēšanas konfigurāciju, datu glabātuves, konteineru reģistru, atkārtoto procesu un asinhrono procesu infrastruktūras uzstādīšanu. Infrastruktūras izveides procesu nepieciešams dokumentēt, lai to varētu uzturēt un pilnveidot atbilstoši mainīgām biznesa prasībām.

Izšķir divu veidu infrastruktūras - tradicionālo un mākoņa bāzēto. Tradicionālo IT infrastruktūru veido iekārtas un programmatūras komponentes: telpas, datu centri, serveri, tīkla aparatūra, datori un lietojumprogrammatūras risinājumi. Tradicionālā infrastruktūra parasti tiek uzstādīta tikai uzņēmuma vai privātai lietošanai. Tradicionālās infrastruktūras izveidei ir nepieciešams vairāk resursu, fiziskās vietas un līdzekļu kā mākoņa bāzētajai (IBM, 2022).

Mākoņa bāzēta IT infrastruktūra ir līdzīga tradicionālajai infrastruktūrai. Lietotāji var piekļūt infrastruktūrai tiešsaistē un izmantot skaitļošanas resursus, neuzstādot iekārtas lokāli, bet izmantojot virtualizāciju. Virtualizācija savieno fiziskus serverus, ko uztur pakalpojumu sniedzējs vienā vai vairākās ģeogrāfiskās vietās. Pakalpojumu sniedzējs sadala un abstrahē resursus, lai tie būtu pieejami gandrīz visur, kur var izveidot interneta savienojumu (IBM, 2022).

Lai nodrošinātu elastību vēlams pielietot mākoņa bāzēto infrastruktūru, iegādāties, ierīkot un uzturēt tradicionālu infrastruktūru un iekārtas, it īpaši globālām sistēmām, ir ekonomisks izaicinājums. Neatbilstošas infrastruktūras izveide radīs sarežģījumus gadījumos, kad zināšanu izguves sistēmai iteratīvi mainīsies prasības un esošā infrastruktūra neiekļaus jaunajām prasībām nepieciešamos resursus. Piemēram, ja izveidota

tradicionālā infrastruktūra un zināšanu sistēmas attīstības iterācijā nepieciešams nodrošināt atbalstu jaunam tirgum citā ģeogrāfiskā lokācijā.

3.1.4. PĀRVALDES UN PUBLISKO LIETOTŅU IZSTRĀDE

Zināšanu izguves sistēmas veido komplekss lietojumprogrammu kopums un bieži vienas organizācijas iekšienē pastāv gan privātas, gan publiskas lietojumprogrammas. Lai nodrošinātu lietojumprogrammu elastību un neatkarību no datu avotiem, tās nepieciešams izstrādāt izmantojot no lietojumprogrammām neatkarīgus datu avotus un API bāzētu izstrādi.

API bāzēta izstrāde ir pieeja, kurā vispirms tiek izstrādātas un veidotas API, pēc tam ap tām tiek izveidota pārējā lietojumprogrammatūra. API bāzētas izstrādes ieguvumi ir sekojošie (Tozzi, 2020):

- visa funkcionalitāte pieejama izmantojot API - kad API ir visas programmatūras arhitektūras un datu piegādes stratēģijas pamats, visas lietojumprogrammu komponentes tiek izstrādātas saderīgas;
- modulāras arhitektūras atbalsts - API bāzēta izstrāde atbalsta modulāras arhitektūras. Pielietojot API var savienot atšķirīgas lietojumprogrammas un atvieglot lietojumprogrammas sadalīšanu moduļos un šos moduļus izvietot izkliedētā infrastruktūrā;
- modulāra programmatūras laidienu nepārtraukta piegāde un vienumu integrēšana (CI/CD) - pielietojot API, lai sadalītu lietojumprogrammas atsevišķos moduļos, ir iespējams nepārtraukti piegādāt katru moduli atsevišķi. Katra lietojumprogrammas daļa (un lietojumprogramma) ir saderīga ar izveidoto API saskarni un arhitektūru, tādēļ viena lietojumprogrammas moduļa problēma neradīs aizkavēšanos pārējā laidienu piegādes procesā. Katru moduli var atjaunināt atsevišķi un atšķirīgā laikā, vienlaikus saglabājot savietojamību ar citiem moduļiem;
- centrāls lietojumprogrammu piegādes punkts - IT organizācijas saskaras ar faktu, ka programmatūras piegādei ir nepieciešamas vairākas komandas - izstrādātāji, sistēmu administratori, programmatūras testētāji un citas, pielietojot API bāzētu izstrādi ir iespējams centralizēt lietojumprogrammu laidienu piegādes procesu.

API bāzēta izstrāde nodrošina arī abstrahēšanos no strikti definētām programmatūras izstrādes valodām, jo katra lietojumprogramma var tikt izstrādāta ar piemērotāko programmēšanas valodu, piemēram, datu apstrādes lietojumprogrammai lietderīgāk pielietot Python programmēšanas valodu, savukārt lietotāju saskarnes izstrādei,

piemēram, Typescript. Lai nodrošinātu elastību ir būtiski izmantot API bāzētu izstrādes pieeju, tādā veidā izveidojot iespējas pielietot sistēmu arī ieinteresētajām trešajām pusēm, piemēram, datu izguvei vai integrācijai ar trešo pušu sistēmām.

Lietojumprogrammu dalīšana pārvaldes un publiskas piekļuves grupās nodrošina iespēju aizsargāt resursus ieviešot stingrākus autentifikācijas un autorizācijas mehānismus privātajām lietojumprogrammām un limitējot pieejamās datu kopas publiskajām lietojumprogrammām. Pārvaldes lietojumprogrammas ar lielākām pieejas tiesībām bieži satur paplašinātu funkcionalitāti, kuras publiska pieejamība var apdraudēt datu nepārtrauktību, integritāti un pieejamību.

3.1.5. MAŠĪNMĀCĪŠANĀS IESPĒJU PIELIETOŠANA

Mašīnmācīšanās pilnveido zināšanu izguvi no lietotāju reakcijas datiem. Datu analīze, vizualizācija un mašīnmācīšanās algoritmu pielietošana rada pievienoto vērtību un atbalsta nākotnes tendenču prognozēšanu un vispārējās situācijas analīzi (vairāk skat., XII pielikumu). Mašīnmācīšanās slāņa izdalīšana nodrošina iespēju abstrahēties no tehnoloģiju un lietojumprogrammu specifiskām prasībām un funkcionalitātes.

Mašīnmācīšanās iespējas var tikt aprobētas izmantojot API bāzētu izstrādi, līdzīgi kā lietojumprogrammu izstrādē, apmācītos modeļus uzstādot kā servisu, kurus iespējams izmantot pēc pieprasījuma. Modeļu ātrdarbība ir nozīmīgs izaicinājums, kuru var risināt lietojot mākoņa bāzētu infrastruktūru un konfigurējot vertikāli un horizontāli automātiski mērogojamas skaitļošanas jaudas.

Mašīnmācīšanās kā atsevišķa slāņa izdalīšana sniedz priekšrocības izstrādāt un ieviest modeļus neatkarīgi no infrastruktūras un lietojumprogrammām un pielietotā zināšanu izguves procesa modeļa. Mašīnmācīšanās slāņa izdalīšana sniedz iespēju izveidot apmācīto modeļu reģistru, kurus var pielietot publiskajās un privātajās lietojumprogrammās izmantojot API vārtejas.

Uzņēmumiem un organizācijām biznesa lēmumu pieņemšanas procesā ir nepieciešama datu balstīta lēmumu pieņemšana. Lai nodrošinātu datu balstītu lēmumu pieņemšanu, uzņēmumiem un organizācijām nepieciešams pielietot informācijas tehnoloģijas un programmatūru. Zināšanu izguves sistēmu izstrādei nepieciešamo tehnoloģiju kopums padara izstrādes procesu sarežģītu un laikietilpīgu, kuru atvieglo zināšanu izguves ietvari. Darba autors izstrādājis konceptuālu zināšanu izguves ietvaru, kuru iespējams pielietot, izstrādājot elastīgus un mērogojamus risinājumus zināšanu izguvei no viedo sistēmu lietotāju reakcijas datiem.

3.2. IETVARS

Ietvars ir abstrakcija, kas apraksta vispārīgu komponentu izkārtojumu, kuru var selektīvi mainīt atbilstoši lietotāja vajadzībām, tādējādi nodrošinot specifisku darbību. Programmatūras ietvari var nodrošināt standarta pieeju lietojumprogrammu izveidei un tiem var būt universālas, atkārtoti lietojamas programmatūras vides, kas nodrošina funkcionalitāti kā daļu no lielākas programmatūras platformas. Programmatūras ietvari var iekļaut atbalsta programmas, kompilatorus, kodu bibliotēkas, rīku kopas un lietojumprogrammu saskarnes, nodrošinot komponentu apvienošanu un projekta vai sistēmas attīstību. Zināšanu izguves sistēmas savukārt veido vairākas savstarpēji savienotas programmatūras sistēmas, tādējādi palielinot sarežģītību. Zināšanu izguves sistēmu izstrādei pieejamo konceptuālo ietvaru skaits ir ierobežots un tie pielietojami kādā noteiktā sfērā, kas bieži nav tiešā veidā pārnesami starp dažādām nozarēm (skat., pirmajā darba daļā).

Tā, kā pieejamie zināšanu izguves ietvari nav tiešā veidā pārnesami starp dažādām nozarēm, autors izstrādājis konceptuālu ietvaru, kuru iespējams aprobēt dažādās nozarēs - UIS-KDF (Jansevskis & Osis, 2024). UIS-KDF dalīts piecos loģiskajos blokos (slāņos) - publisko lietotņu, pārvaldes lietotņu, MM, ietekmējošo kritēriju un tehnoloģiju:

- tehnoloģiju slānis - horizontāli un vertikāli mērogojams, ietver tehnoloģiju stekus, operētājsistēmas, konteinerus, procesus, sakaru protokolus, datubāzes, vārtejas, uzraudzības un pārvaldes risinājumus;
- publisko lietotņu slānis - nodrošina biznesa funkcionalitātes prasības atbilstoši gala patērētāju vēlmēm;
- pārvaldes lietotņu slānis - iekšējo procesu un procedūru pārvaldes lietotņu slānis, kura uzdevums ir nodrošināt pārvaldes un vadības funkcionalitāti atbilstoši uzņēmuma biznesa prasībām;
- MM slānis - nodrošina zināšanu izguves funkcionalitāti no lietotāju reakcijas datiem un zināšanu izguves modeļu vadību.
- ietekmējošo kritēriju - apkopo zināšanu izguves sistēmu regulējošos tiesiskos aspektus un spēkā esošās regulas, kā arī kibernetikas faktorus (detalizēti skat., darba otrajā daļā).

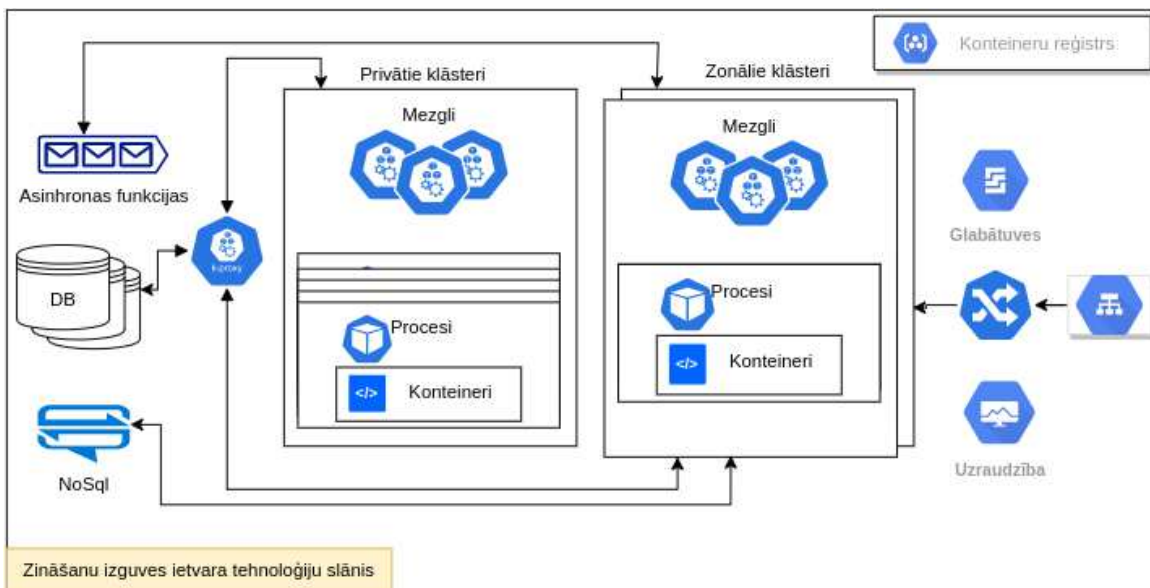
3.2.1. TEHNOLOĢIJU SLĀNIS

Tehnoloģijas un infrastruktūra nodrošina zināšanu izguves sistēmu funkcionalitāti, atbalsta biznesa procesus un palīdz veikt datus balstītu lēmumu pieņemšanu. Tehnoloģiju

slāņa izdalīšana nodrošina tehnoloģiju atdalīšanu no biznesa, lietojumprogrammu un mašīnmācīšanās funkcionalitātes (skat. 3.2.att.). Tehnoloģiju slānis izstrādāts paredzot mākoņskaitļošanas tehnoloģiju pielietošanu un to nodrošinātās priekšrocības.

Mākoņskaitļošanas tehnoloģiju priekšrocības ir dalītās skaitļošanas potenciāls, izklīdētās datubāzes un virtualizācijas tehnoloģijas, kā arī automatizēta vertikālā un horizontālā mērogojamība. UIS-KDF ietvara tehnoloģiju slānim nepieciešams pielietot mākoņskaitļošanas pakalpojumus, piemēram, Amazon Web Services, Microsoft Azure, Google Cloud Platform. Uzskaitītie pakalpojumu sniedzēji spēj piedāvāt infrastruktūru, kura mērogojama atbilstoši izstrādes prasībām un spēj nodrošināt pilnu zināšanu izguves sistēmas izveides un datu apstrādes ciklu.

Ieteicamais scenārijs, lai izveidotu elastīgu tehnoloģiju slāni, ir pielietot konteinerus, piemēram, Docker (*Docker*, 2022), LXD (*LXD*, 2022), Containerd (*Containerd*, 2022), konteineru orķestrēšanas platformas un klasteru tehnoloģijas, piemēram, Kubernetes (*Kubernetes*, 2022), Rancher (*Rancher*, 2022), Google Cloud Run (*Google Cloud Run*, 2022), AWS Fargate (*AWS Fargate*, 2022). 3.2.attēlā attēlots tehnoloģiju slānis pielietojot Kubernetes, Docker, Cloud Run un Containerd risinājumus.



3.2.attēls. Zināšanu izguves ietvara tehnoloģiju slānis (autora veidots attēls).

Mākoņskaitļošanas platformā izvietota datu glabātuve nodrošina tādu funkcionalitāti kā datu nepārtrauktība, integritāte un pieejamība (skat., 3.2.att.). Mākoņglabātuvju servisa pakalpojumu izvēle atkarīga no tehniskajām prasībām, un šos

pakalpojumus ir iespējams izvietot kādā no pakalpojumu nodrošinātāju infrastruktūrām. Mākoņskaitļošanas platformas uzdevums UIS-KDF ietvarā ir nodrošināt lietojumprogrammu saskarni, kuru var pielietot privātās, publiskās un MM slāņu lietotnes.

Datu apjomi, ko spēj saražot viedās sistēmas un ar tām savienotās IoT iekārtas, ir apjomīgi. Tādēļ nozīmīgs uzdevums ir izvēlēties tādu infrastruktūru, kuru ir iespējams salīdzinoši vienkārši vertikāli un horizontāli mērogot. Liela apjoma datu uzglabāšanu var realizēt lietojot datubāzu risinājumus un datubāzu klāsterus, kā, piemēram, tradicionālās relāciju bāzētās datubāzu sistēmas (SQL) (skat., 3.2.att.) vai jaunā veida glabātuves (NoSQL), kā piemēri minami Apache Flume, HDFS, SQL, MongoDB u.c. (skat., 2.3.att.). Atsevišķi izdalāmi gadījumi, kuros M2M savstarpējā komunikācija var būt atvienota no globālā tīmekļa, kas rada nepieciešamību datus glabāt uz vietas iekārtā. Lai šo lietošanas gadījumu risinātu, nepieciešams realizēt scenāriju, kurā dati tiek glabāti iekārtās līdz mirklim, kad tās ir iespējams sinhronizēt ar tīmekļa datubāzēm tālākai lietošanai.

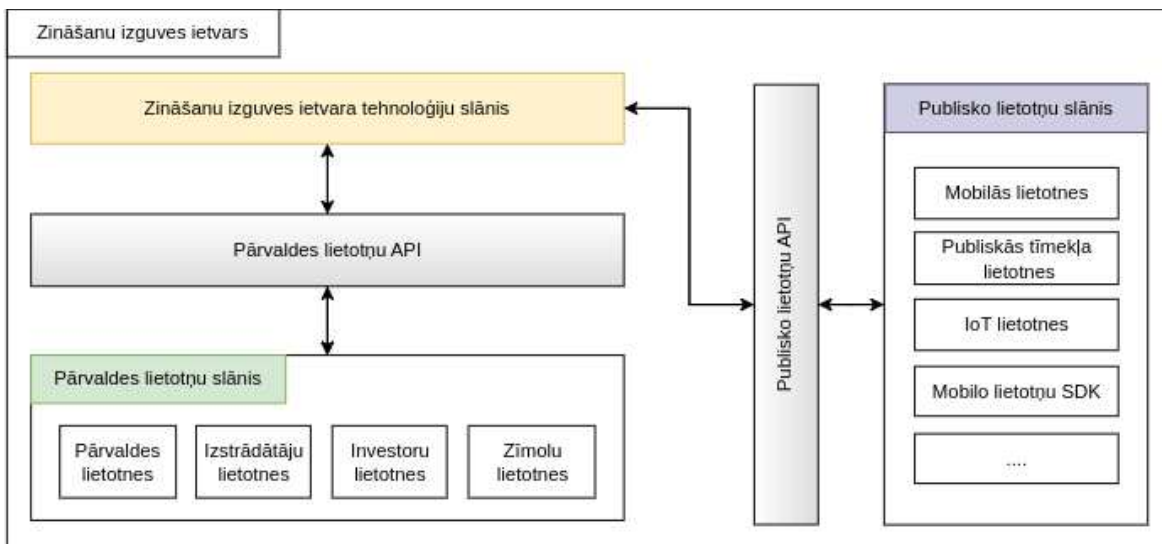
Datu glabāšanai M2M scenārijos nepieciešams izmantot vienkāršota veida datubāzu tehnoloģijas, kuras izmanto mazāk energoresursu un kuras iespējams glabāt pievienotajās iekārtās vai iekārtu klāsteros (IoT iekārtās). Potenciāli pielietojami ir datņu bāzēti datubāzu tehnoloģiju risinājumi, kā iespējamus risinājumus var minēt RealmDB, SQLite, ORMLite un Berkeley DB (*Oracle Berkeley DB*, 2022; *OrmLite*, 2022; *Realm Database*, 2022; *SQLite*, 2022). Bieži vien dati, kurus glabā IoT iekārtas būs telpiski, tādēļ var būt nepieciešams pielietot datubāzu versiju, kura atbalsta ģeotelpisko datu uzglabāšanu, kā, piemēram, SpatiaLite (*SpatiaLite*, 2022).

Tehnoloģiju slānis iekļauj arī slodzes balansēšanas vadību, asinhrono darbību pārvaldi, uzraudzības risinājumus, sistēmu attēlu reģistrus, publisko un pārvaldes lietotņu API, kā arī atbalstošās tehnoloģijas. Tehnoloģiju slānis UIS-KDF ietvarā nodrošina tehnoloģiju atdalīšanu no biznesa, lietojumprogrammu un mašīnmācīšanās funkcionalitātes. Tehnoloģiju uzturēšanas uzdevumi bieži vien būtiski atšķiras no organizācijas biznesa mērķiem, tādēļ funkcionalitāti nodrošinošās tehnoloģijas izdalītas atsevišķā slānī.

3.2.2. PUBLISKO UN PĀRVALDES LIETOTŅU SLĀŅI

Publisko lietotņu slānis (skat., 3.3.att.) UIS-KDF ietvarā nodrošina biznesa funkcionalitātes prasības atbilstoši gala patērētāju vēlmēm. Publisko lietotņu slānis strādā ar platformas tehnoloģiju slāņa API saskarnēm, nodrošinot elastību publiskās lietotnes pārveidot, pielāgot, pievienot jaunas un aizstāt.

Pārvaldes lietotņu slānis (skat., 3.3.att.) satur uzņēmuma iekšējo procesu un procedūru pārvaldes lietotņu kopu, kuru uzdevums ir nodrošināt pārvaldes un vadības funkcionalitāti atbilstoši uzņēmuma biznesa prasībām. Nozīmīgākās atšķirības starp publisko un pārvaldes lietotņu slāņiem ir pieejas tiesību vadība un mērķauditorija.



3.3.attēls. Publisko un pārvaldes lietotņu slāņi (autora veidots attēls).

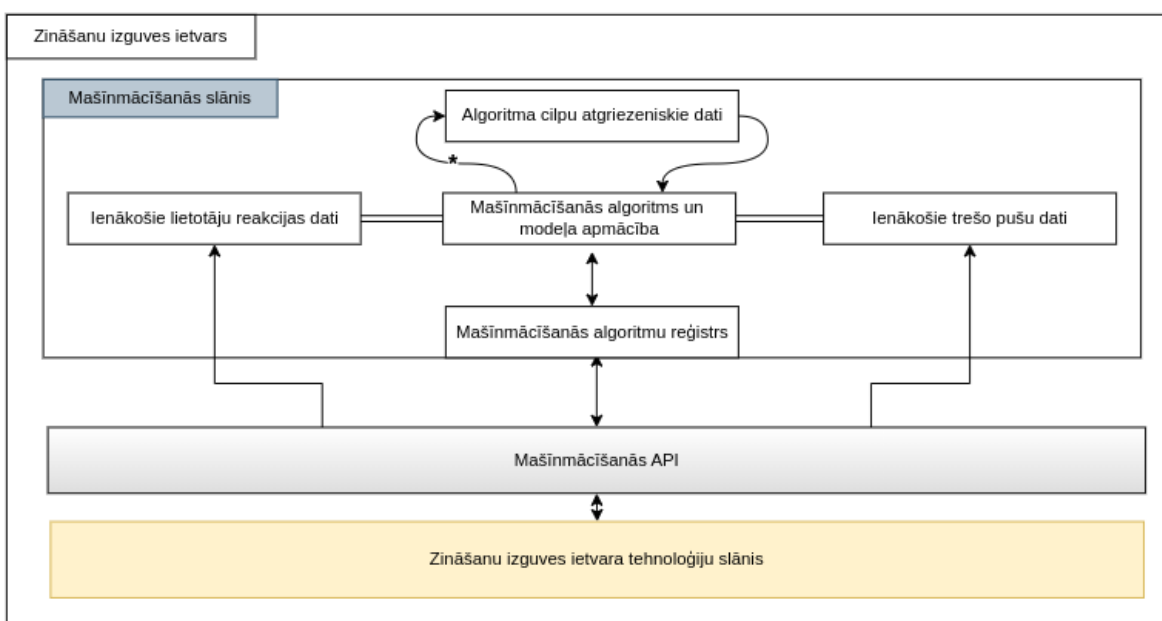
Gan publisko, gan pārvaldes lietotņu slāņi iekļauj visu platformu un sistēmu tīmekļa bāzētas, mobilo iekārtu, hibrīdas un darbstaciju lietotnes, kā arī gan atvērtā pirmkoda, gan slēgtas piekļuves pirmkoda lietotnes (skat., 3.3.att.). Būtisks nosacījums šīm lietotnēm ir spēja darboties ar attālinātu lietojumprogrammu saskarni neatkarīgi no API tehnoloģijas veida (REST, GraphQL, RPC, SOAP). Savukārt lietotnes var tikt izstrādātas ar jebkuru piemērotu izstrādes valodu atbilstoši konkrētajiem lietošanas gadījumiem.

Publisko un pārvaldes lietotņu būtiska atšķirība ir gala lietotāji un potenciālās slodzes, kuras spēj radīt noteiktās lietotnes. Pārvaldes lietojumprogrammu lietotāju skaits vienmēr būs ierobežots un kontrolējams, savukārt publisko lietojumprogrammu lietotāju skaits var nebūt limitēts. Publisko lietotņu API izstrādes procesā nepieciešams ņemt vērā mērogojamību, slodzes vadību un striktas autentifikācijas metodes.

Uzņēmumu un organizāciju biznesa mērķi ir vērsti uz pakalpojumu nodrošināšanu gala patērētājiem, un tās nodrošina publiskās lietotnes. Savukārt pārvaldes lietotnes paredzētas uzņēmuma iekšējai lietošanai. Tādēļ pārvaldes un publisko lietotņu funkcionalitāti ir nepieciešams atdalīt.

3.2.3. MAŠĪNMĀCĪŠANĀS SLĀNIS

Mašīnmācīšanās slānis (skat., 3.4.att.) nodrošina zināšanu izguves funkcionalitāti no lietotāju reakcijas datiem un zināšanu izguves modeļu pārvaldi. Būtiskie nosacījumi slānim ir nodrošināt MM modeļu iespējas un izvietojumu. Zināšanu izguves sistēmu nolūks ir nodrošināt datus balstītu lēmumu pieņemšanu un zināšanu izguvi no datiem. Minētos uzdevumus pilnveido mašīnmācīšanās, kas nodrošina iespēju zināšanu izguves sistēmām ar noteiktu precizitāti prognozēt rezultātus kā arī pilnveidot sniegtos pakalpojumus, piemēram, nodrošinot sarunbotu funkcionalitāti.



3.4.attēls. MM slānis (autora veidots attēls).

Mašīnmācīšanās slānis iekļauj MM algoritmu reģistru, algoritmu cilpu atgriezeniskos datus, trešo pušu datus un lietotāju reakcijas datus (skat.. 3.4.att.). MM slānis iekļauj MM algoritmu reģistru, nodrošinot iespēju pielietot vairākus MM algoritmus atšķirīgiem nolūkiem. MM reģistrs nodrošina iespēju noteiktu algoritmu aizvietot, ja ir tāda nepieciešamība kā arī tos izvietot vienotā sistēmā. Izdalot MM slāni zināšanu izguves sistēmai var nošķirt mašīnmācīšanās funkcionalitāti un apmācības procesu no publisko un privāto lietotāju funkcionalitātes.

Eksistējošo mašīnmācīšanās algoritmu sarežģītība, izaicinājumi darboties ar mainīgām ieejas datu kopām, hiperparametru identificēšana, mācīšanās algoritmu pielāgošana, ierobežojumi mērogošanā un paralēlajā apstrādē, izvirza prasības, kuras jāņem vērā ieviešot mašīnmācīšanās algoritmus zināšanu izguves sistēmu izstrādē.

Mašīnmācīšanās algoritmu modeļus bieži izstrādā ņemot vērā aplūkojamās sfēras nozīmīgākos faktorus un izmantojot pieejamos vēsturiskos lietotāju reakcijas datus. Pielietojot mašīnmācīšanās tehnoloģijas var izstrādāt algoritmus, kuri spēj izgūt zināšanas no liela apjoma lietotāju reakcijas datiem. Mašīnmācīšanās pakāpeniski ir ieņēmusi būtisku lomu dažādās sociālās sfērās, piemēram, valodas apstrādē, dabīgās valodas izpratnē, neirozinātnē, lietu internetā u.c (Zhou et al., 2017).

MM rada izaicinājumus arī uzņēmumu speciālistu piesaistīšanā, jo gandrīz neeksistē eksperti, kuriem būtu visas nepieciešamās zināšanas par MM, datu apstrādes pieejas tehnoloģijām, statistisko analīzi, pētāmo nozari, datubāzu tehnoloģijām, tehnoloģiju konfigurāciju, programmēšanas valodām un lietotņu izstrādi. Pakāpeniski notiek mēģinājumi izstrādāt informācijas sistēmas, kuras ir vieglāk saprast un lietot nespeciālistiem (Zhou et al., 2017). Lai risinātu šos izaicinājumus zināšanu izguves sistēmām jāizdala MM slāni, tādējādi specifisko funkcionalitāti un uzņēmumu speciālistu prasības nodalot no kopējās zināšanu izguves sistēmas darbības.

Pielietojot mašīnmācīšanos zināšanu izguves sistēmās, jāapsver iespēja padarīt risinājumus mērogojamus, atkārtoti un paralēli izpildāmus, konfigurējamus un salīdzinoši ērti lietojamus. Šim uzdevumam atbilstošus risinājumus piedāvā, piemēram, TensorFlow (*TensorFlow*, 2022) un Scikit Learn (*Scikit Learn*, 2022) platformas, kuras ir atvērtā koda bibliotēkas skaitļošanas operāciju veikšanai, izvietojamas uz vairākiem procesoriem vai grafiskajām kartēm.

Datu apstrādes darbībām jānodrošina pievienotā vērtība un pietiekami ātra datu apstrāde. Datu analīze un MM algoritmu pielietošana rada pievienoto vērtību, kas sniedz atbalstu nākotnes tendenču prognozēšanā un situācijas izpratnē. Mašīnmācīšanās slāņa izdalīšana nodrošina iespēju abstrahēties no tehnoloģiju un lietojumprogrammu specifiskām prasībām un funkcionalitātes.

3.2.4. UIS-KDF

Zināšanu izguve attiecas uz zināšanu izguves procesu datus un akcentē konkrētu datu izguves metožu pielietojumu. Zināšanu izguves mērķis ir izgūt zināšanas no datiem. Zināšanas tiek izgūtas, izmantojot datu izguves algoritmus, lai saskaņā ar definētiem nosacījumiem un specifikācijām izgūtu zināšanas no datiem (Technopedia Inc, 2017). Zināšanu izguves procesu apraksta arī vairāki procesu modeļi, kas izstrādāti gan akadēmiskajā, gan nozares vidē (skat., 1.3.daļa). Zināšanu izguves procesu apraksta kā

vairākas pamata aktivitātes, kuras iekļautas vispārējos projektu vadības pamatprincipos (Osei-Bryson & Barclay, 2015):

1. skaidras izpratnes veidošana par organizācijas mērķiem un uzdevumiem;
2. saistošo datu identificēšana, analīze un sagatavošanai;
3. algoritmu pielietošana, lai no datiem iegūtu saistošas zināšanas;
4. piemērotāko modeļu izvērtēšana un atlase;
5. iegūto zināšanu pielietošana un izmantošana.

Zināšanu izguves procesu ieviešana uzņēmumu darbībā saistīta ar vairākām problēmsituācijām. Daļa zināšanu izguves algoritmu un rīku apstājas pie modeļu izveides un piegādes, kas apmierina tehniskās prasības. Zināšanu izguves modeļi tiek izstrādāti un uzņēmuma iekšienē ne vienmēr ir zināšanas un izpratne kā modeļus pielietot biznesa lēmumu pieņemšanā (Cao et al., 2010). Praktisko zināšanu atklāšana ir slēgts problēmu risināšanas process, kas ietver virkni mērķtiecīgu darbību: problēmas definēšanu, ietvara/modeļa izstrādi un paredzēt, lai nodrošinātu funkcionējošus biznesa noteikumus, kurus var nemanāmi saistīt vai integrēt ar biznesa procesiem un sistēmām (Cao et al., 2010). Savukārt IT sistēmās joprojām ir vērojama pieaugoša tendence informācijas un slēpto korelāciju iegūšanai izmantot pieejamos datus. Būtiskākie faktori, kas to veicinājuši ir vēlme sniegt labākus pakalpojumus lietotājiem (Osman, 2019).

Lai uzņēmumi sniegtu labākus pakalpojumus lietotājam, ir nepieciešams veids kā izgūt zināšanas no lietotāju reakcijas datiem un tās var izgūt izmantojot zināšanu izguves sistēmas. Zināšanu izguves sistēmu izstrāde ir sarežģīts un laikietilpīgs process, kuru atvieglo zināšanu izguves ietvari, kas nodrošina izstrādes vadlīnijas (skat., 1.4.daļa). Zināšanu izguves ietvari no lietotāju reakcijas datiem tiek pielietoti zināšanu izguves sistēmu arhitektūras agrā izstrādes stadijā.

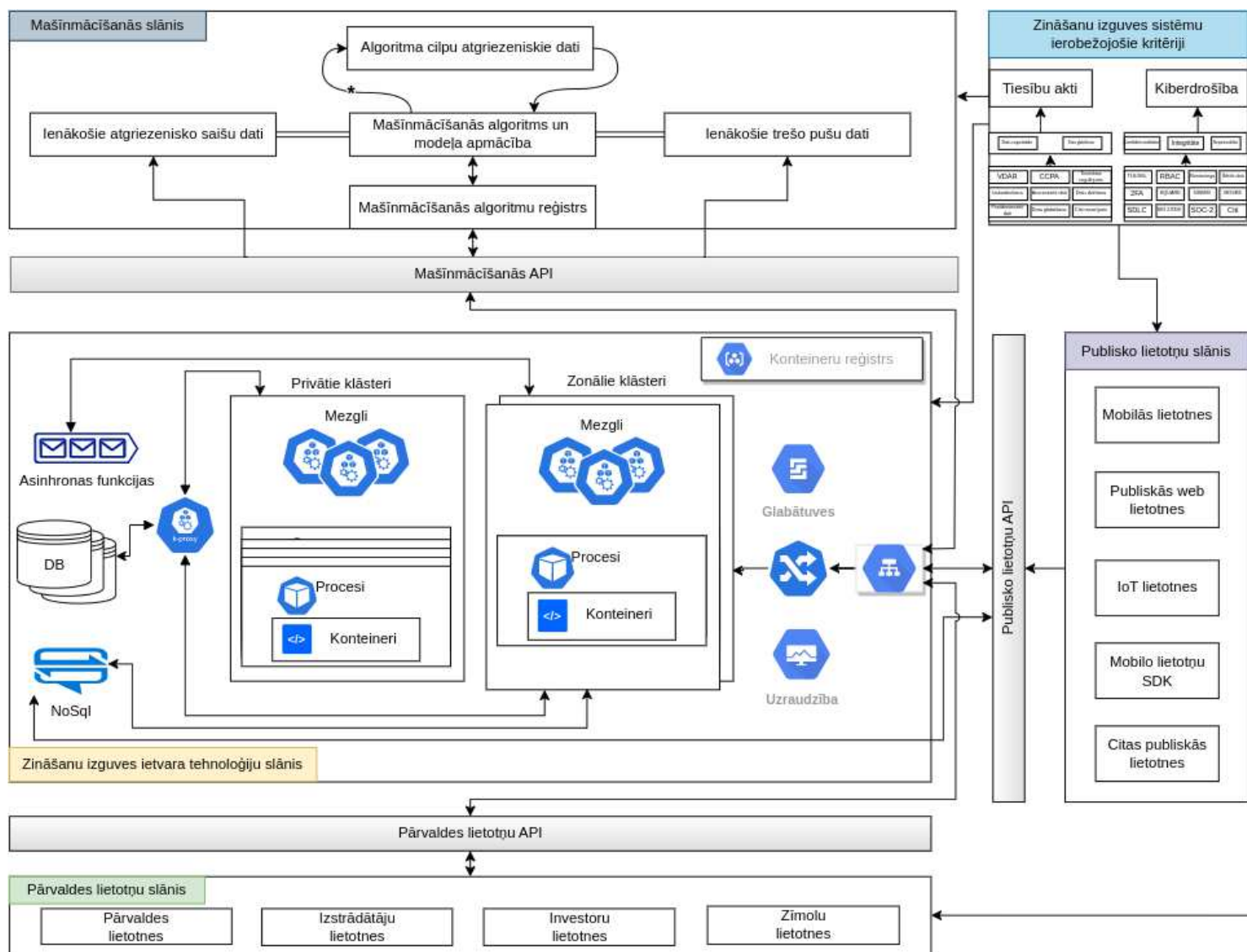
Darba autora izstrādātais UIS-KDF ietvars (skat., 3.5.att.) pielietojams izstrādājot elastīgus un mērogojamus risinājumus zināšanu izguvei no viedo sistēmu lietotāju reakcijas datiem. UIS-KDF ietvars dalīts piecos slāņos - publisko lietotņu, pārvaldes lietotņu, MM un tehnoloģiju (slāņi detalizēti 3.2.1., 3.2.2., 3.2.3. apakšnodaļās un darba otrajā daļā):

- tehnoloģiju slānis - ietver zināšanu sistēmas nodrošināšanai nepieciešamās tehnoloģijas. Būtiski nosacījumi slānim ir iespēja dinamiski palielināt un samazināt skaitļošanas jaudas atbilstoši nepieciešamajām, kā arī vairāku līmeņu lietotāju autentifikāciju, pilnu audita žurnālu uzglabāšanu, ierobežotas pieejas tiesības atbildīgajām personām;

- publisko lietotņu slānis - strādā ar tehnoloģiju slāņa API saskarnēm, nodrošinot elastību tās pārveidot, pielāgot, pievienot jaunas un aizstāt;
- pārvaldes lietotņu slānis - nodrošina pārvaldes un vadības funkcionalitāti atbilstoši uzņēmuma biznesa prasībām;
- MM slānis - nodrošina zināšanu izguves funkcionalitāti no lietotāju reakcijas datiem, zināšanu izguves modeļu pārvaldi un MM modeļu potenciālu un izvietošanu.
- Ietekmējošo kritēriju - apkopo zināšanu izguves sistēmu regulējošos tiesiskos aspektus un spēkā esošās regulas, kā arī kiberdrošības faktoros (detalizēti skat., darba otrajā daļā).

UIS-KDF ietvara dalījums slāņos (skat., 3.5.att.) nodrošina elastību atbilstoši mainīgiem biznesa mērķiem. UIS-KDF ietvarā tehnoloģijas, pārvaldes un publiskās lietotnes un mašīnmācīšanās izdalīti kā savstarpēji vāji saistīti slāņi. Tehnoloģiju un infrastruktūras prasības izstrādes procesos būtiski atšķiras no prasībām pārvaldes un publiskajām lietotnēm tāpat kā mašīnmācīšanās procesiem. Savukārt uzņēmumiem datus balstītu lēmumu pieņemšanas procesam nozīmīga ir visu minēto komponentu savstarpēja mijiedarbība.

UIS-KDF ietvars paredzēts zināšanu izguves sistēmu plānošanas agrai stadijai - infrastruktūras plānošanas atbalsta procesam par tehnoloģijām, to arhitektūru un plānošanu atbildīgajiem uzņēmumu darbiniekiem. UIS-KDF ietvara nolūks ir atvieglot zināšanu izguves sistēmu izstrādes sarežģīto un laikietilpīgo procesu, uzrādot arhitektūras plānošanas un organizācijas vadlīnijas (Jansevskis & Osis, 2024a).



3.5.attēls. UIS-KDF ietvars (autora veidots).

UIS-KDF ietvara pazīmes apkopotas, pielietojot 1.4.4. apakšnodaļā (skat., 1.3.tabulu) izstrādāto zināšanu izguves ietvaru pazīmju kopumu un uzrādītas 3.1.tabulā.

3.1. tabula. UIS-KDF ietvara pazīmes (autora veidota)

Pazīme	UIS-KDF ietvars	Apraksts
Modulārs	✓	Dalīts piecos slāņos - publisko lietotņu, pārvaldes lietotņu, MM, platformas tehnoloģiju un ierobežojošo faktoru.
Izstrādātāji abstrahēti no datu apstrādes platformu sarežģītības	✓	Datu uzglabāšanas un datu apstrādes platformas atdalītas no lietojumprogrammu izstrādes. Izstrādātāji var koncentrēties lietotņu izstrādes procesiem.
Dalīta infrastruktūra	✓	Horizontāli un vertikāli mērogojama, ietver tehnoloģiju stekus, operētājsistēmas, konteinerus, procesus, sakaru protokolus, datubāzes, vārtejas, uzraudzības un pārvaldes risinājumus.
Datu priekšapstrādes modulis	✓	Iekļauts platformas tehnoloģiju slānī.
Rezultātu apstrādes un izplatīšanas modulis	✓	Nodrošināts pielietojot lietojumprogrammu saskarni un trešo pušu savietojamas lietotnes.
Savienojami programmēšanas mezgli	✓	Realizācijas scenārijs - pielietojot lietojumprogrammu saskarnes.
Augstas veiktspējas mērogojama infrastruktūra	✓	Realizācijas scenārijs - automātiski mērogojamas infrastruktūras pielietošana.
Zināšanu izguves atbalsts	✓	Iekļauts MM slānī, kurā iespējams ieviest un pievienot jaunus modeļus atkarībā no uzņēmuma biznesa mērķiem.
Lielo datu platformu atbalsts	✓	Iekļauts platformas tehnoloģiju slānī.
Tiešsaistes analītikas modulis	✓	Iekļauts platformas nodrošinošo tehnoloģiju slānī.

Pazīme	UIS-KDF ietvars	Apraksts
Drošība un tiesību aktu ierobežojumi kā daļa no ietvara	✓	Iekļauts kā ietvara ietekmējošo kritēriju slānis
Mērķauditorija	✓	Uzņēmumu un organizāciju vadošie tehniskie darbinieki, kuru uzdevums ir atbalstīt datus balstītu lēmumu pieņemšanu. Vienlaikus nodrošinot tehnoloģiju elastību un potenciālas biznesa mērķu maiņas, kā arī atbalstot strauju izaugsmi.

Arvien pieaugošais tehnoloģiju izmaiņu temps, kā arī vienkāršas un elastīgas prakses pieņemšana, kas nodrošina ātrus piegādes ciklus, mācīšanos un uz datiem balstītus lēmumus, ir padarījusi tradicionālo tehnoloģiju izstrādi par smagnēju procesu. Termini bieži tiek kavēti, tehnoloģijas kļūst novecojušas, prioritātes un klientu vēlmes mainās un konkurence pieaug. Rezultātā organizācijas arvien biežāk nokļūst starp izvēli ieviest izmaiņas vai izpildīt sākotnējo plānu.

Tabulā (skat., 3.2. tabulu) apkopotas kvantitatīvo rādītāju pazīmes, kuras ieteicams pielietot, lai novērtētu UIS-KDF ietvara aprobācijas lietderīgumu. Novērtējot zināšanu izguves ietvara efektivitāti, kvantitatīvie mērījumi var sniegt skaidrāku un objektīvāku ieskatu. UIS-KDF ietvara pielietojuma kvantitatīvo rādītāju novērtējuma pieejas mērījumu saraksts (skat., 3.2. tabulu) UIS-KDF aprobācijas novērtēšanai nav galīgs, un, to nepieciešams pielāgot noteiktam aprobācijas scenārijam, lai tas atbilstu dažādu organizāciju vajadzībām un mērķiem. Tādi rādītāji kā sistēmas darbības laiks, datu integrācijas ātrums un modeļu precizitāte sniedz lietderīgu ieskatu ietvara veiktspējā, šo mērījumu atbilstība un nozīme dažādās nozarēs un lietošanas gadījumos var atšķirties. Piemēram, veselības aprūpes organizācija par prioritāti var noteikt datu drošību un prognožu rezultātu uzlabošanu, savukārt reklāmas nozares uzņēmums var vairāk pievērsties datu apstrādes ātrumam reāllaikā un krāpšanas atklāšanas precizitātei. Tādēļ uzņēmumiem un organizācijām ir nozīmīgi novērtēt un pielāgot rādītājus, lai nodrošinātu, ka tie atbilst darbības mērķiem un normatīvajām prasībām.

Daži no ieteiktajiem mērījumiem aprobācijas scenārijā var būt subjektīvi un atkarīgi no konteksta. Lietotāju apmierinātības rādītājus, uzņēmējdarbības ietekmes rādītājus un lēmumu pieņemšanas uzlabojumu procentus ietekmē organizācijas kultūra, ieinteresēto pušu vēlmes un konkrētais konteksts, kurā tiek izmantota zināšanu izguves

sistēma. Šiem rādītājiem nepieciešami kvalitatīvi novērtējumi un ieinteresēto pušu ieguldījums, lai nodrošinātu visaptverošu novērtējumu. Uzņēmumiem un organizācijām izvērtējot UIS-KDF aprobāciju jāapsver iespēja papildināt kvantitatīvos datus ar kvalitatīvu atgriezenisko saiti no lietotājiem un ieinteresētajām personām, lai iegūtu izpratni par UIS-KDF ietekmi. UIS-KDF ietvara pielietojuma kvantitatīvo rādītāju novērtējuma pieeja (skat., 3.2. tabulu) ir uzskatāma kā sagataves piemērs šāda novērtējuma veikšanai, nevis kā galīgi kritēriji, organizācijas var pielāgot savus vērtēšanas procesus, lai atspoguļotu specifiskos apstākļus un stratēģiskos mērķus.

3.2.tabula. UIS-KDF ietvara kvantitatīvo rādītāju novērtējuma pieeja (autora veidota)

Slānis	Metrika	Mērijums
Tehnoloģiju slānis	Datu integrācijas ātrums. <i>Laiks, kas nepieciešams jaunu datu avotu integrēšanai.</i>	Vidējais laiks stundās/dienās (stundas/diena)
	Darbspējas laiks. <i>Procentuālais laika apjoms, kurā sistēma darbojas un ir pieejama.</i>	Procentuālais pieejas laiks (procenti)
	Datu apstrādes ātrums. <i>Vidējais laiks, lai apstrādātu vienu datu vienību.</i>	Laiks vienai datu vienībai (sekundes/minūtes)
Publisko lietotņu slānis	Lietotāju pieņemšanas līmenis. <i>Lietotāju skaits, kuri aktīvi izmanto publiskās lietojumprogrammas.</i>	Lietotāju skaits mēnesī (vienības)
	Lietotāju apmierinātības līmenis. <i>Vidējais apmierinātības vērtējums lietotāju aptaujās.</i>	Vidējais novērtējums (vērtējuma skala)
	Sistēmas atbildes ātrums. <i>Vidējais laiks, kurā sistēma reaģē uz lietotāju vaicājumiem.</i>	Vidējais laiks (milisekundēs/sekundēs)
Privāto lietotņu slānis	Datu drošības incidenti. <i>Ziņoto datu pārkāpumu vai drošības incidentu skaits.</i>	Incidentu skaits (skaitlis)
	Piekļuves laiks. <i>Vidējais laiks, kas nepieciešams datu izgūšanai no privātām lietojumprogrammām.</i>	Vidējais datu ieguves laiks (milisekundes)
	Atbilstības standartiem līmenis. <i>Datu aizsardzības noteikumu</i>	Procenti

Slānis	Metrika	Mērījums
	<i>ievērošanas procentuālā daļa.</i>	
Mašīnmācīšanās slānis	Modeļu precizitāte <i>Sistēmā izmantoto mašīnmācīšanās modeļu vidējā precizitāte.</i>	Procenti
	Apmācības laiks <i>Vidējais laiks, kas nepieciešams mašīnmācīšanās modeļa apmācībai.</i>	Vidējais laiks (stundas/dienas)
	Prognozes latentums <i>Vidējais laiks, kas nepieciešams, lai modeļi atgrieztu prognozes.</i>	Vidējais laiks (milisekundes/sekundes)
Ietekmējošo kritēriju slānis	Lēmumu pieņemšanas uzlabojums. <i>Procentuāli uzlabota lēmumu pieņemšanas precizitāte, pateicoties no sistēmas izgūtajiem ieskatiem.</i>	Procentuālais uzlabojums (procenti)
	Peļņa no kapitāla ieguldījuma. <i>Finansiālā atdeve, ko rada ietvara ieviešana, salīdzinot ar tā izmaksām.</i>	Finansiālais ieguvums (procenti)
	Uzņēmējdarbības ietekmes vērtējums. <i>Rādītājs, kas atspoguļo ietvara kopējo ietekmi uz uzņēmējdarbību, kas aprēķināts, pamatojoties uz konkrētiem uzņēmuma veidotiem veikspējas indeksiem.</i>	Kombinētais mērījums atbilstoši veikspējas indeksam (mērvienība atbilstoši uzņēmuma/organizācijas ieviestajai sistēmai)

Darba autora izstrādāto UIS-KDF ietvaru paredzēts pielietot, izstrādājot elastīgus un mērojamus risinājumus zināšanu izguvei no viedo sistēmu lietotāju reakcijas datiem. Primāri sasniedzamais uzdevums organizācijai ir sniegt pievienoto vērtību tās klientiem. Savukārt izvēlētajām tehnoloģijām jābūt elastīgām, lai tās varētu piegādāt pievienoto vērtību organizācijai un tās klientiem.

3.3. UIS-KDF PIELIETOJUMA METODOLOĢIJA

Izstrādātā UIS-KDF ietvara primārais pielietošanas mērķis ir zināšanu izguves sistēmu arhitektūras izstrāde. Autors zināšanu izguves sistēmu izstrādei noteicis secīgus posmus (skat. 3.1.att.). Izstrādātā UIS-KDF ietvara pielietošana atbilstošajos posmos uzrādīta 3.6.attēlā.

1. Prasību noteikšana (skat., 3.6. att.):

- a. prasību noteikšanas procesā UIS-KDF ietvaru nepieciešams aprobēt noteiktajam lietošanas gadījumam, izveidojot UIS-KDF ietvara aprobāciju, kuras laikā ietvara slāņos tiek iekļauti prasībām atbilstošie risinājumi un tiek izveidots zināšanu izguves sistēmas ietvars;
- b. zināšanu izguves sistēmas arhitektūras izstrādes procesā, definējot zināšanu izguves sistēmas tiesiskās prasības lietojams UIS-KDF ietvara ietekmējošo kritēriju slānis nosakot saistošos tiesiskos normatīvus un kiberdrošības nosacījumus;
- c. infrastruktūras, izmitināšanas pakalpojumu un programmatūras prasību noteikšanas procesā lietojams UIS-KDF ietvara tehnoloģiju slānis definējot sistēmas tehniskās prasības;
- d. zināšanu izguves sistēmas attīstības iterācijās (jaunievedumu un uzlabojumu izstrādes process) zināšanu izguves ietvara aprobācija (pielāgotais UIS-KDF ietvars) izmantojama kā sistēmas arhitektūras shēma, kura sniedz sistēmas kopskatu meta līmenī un uzrāda sistēmas meta līmeņa procesus un datu plūsmas;
- e. UIS-KDF ietvara aprobācija, izmantojot tehnoloģiju slāni, pielietojama definējot funkcionalitātes prasības un zināšanu izguves sistēmas atsevišķu daļu veikspējas un izvietojuma prasības;
- f. sasniedzamais uzdevums: UIS-KDF ietvara aprobācija zināšanu izguves sistēmā.

2. Tehnoloģiju izvēle (skat., 3.6.att.):

- a. posmā pielietojams aprobētais UIS-KDF ietvars un tā tehnoloģiju slānis;
- b. zināšanu izguves sistēmai, atbilstoši pirmajā posmā noteiktajām prasībām, jā sastāda pielietojamo specifisko tehnoloģiju kopumu;
- c. ietvars pielietojams attīstības iterāciju tehnoloģiju izvēles procesā un definējot atsevišķu sistēmas daļu prasības;
- d. sasniedzamais uzdevums: zināšanu izguves sistēma, kurā iekļauti specifiski, funkcionalitātes prasībām atbilstoši, risinājumi.



3.6.attēls. UIS-KDF ietvara pielietošana (autora veidots).

3. Infrastruktūras izveide (skat., 3.6.att.):
 - a. infrastruktūras izveides procesā plānojot un izstrādājot zināšanu izguves sistēmas infrastruktūru lietojams otrajā posmā papildinātais, aprobētais UIS-KDF ietvars, kuram atbilstoši tiek izveidota un izvietota infrastruktūra;
 - b. UIS-KDF ietvars attīstības iterācijās pielietojams plānojot sistēmas daļu infrastruktūras izveidi un uzturēšanu;
 - c. sasniedzamais uzdevums: atbilstoši UIS-KDF ietvaram izveidota un izvietota infrastruktūra, kura nodrošina publisko, pārvaldes lietotņu un MM slāņu funkcionalitātes prasības.
4. Pārvaldes lietotņu izstrāde:
 - a. pārvaldes lietotņu izstrādes procesā aprobētais UIS-KDF ietvars pielietojams lietotņu izstrādes plānošanas procesā nosakot pārvaldes lietotņu pieejas tiesības un pielietojamos (izstrādājamos) programmējamos mezglus;
 - b. attīstības iterācijās ietvars lietojams funkcionalitātes plānošanas procesā;
 - c. sasniedzamais uzdevums: aprobētais UIS-KDF ietvars nodrošina pārvaldes lietotņu izstrādes funkcionalitātes un drošības prasības, pārvaldes lietotnes, kuras nodrošina atbilstošo biznesa funkcionalitāti un datus balstītu lēmumu pieņemšanu.
5. Publisko lietotņu izstrāde:
 - a. aprobētais UIS-KDF ietvars lietojams publisko lietotņu plānošanas un izstrādes procesos nosakot pieejas tiesības un pielietojamos, kā arī izstrādājamos publiskos programmējamos mezglus;
 - b. attīstības iterācijās aprobētais ietvars lietojams funkcionalitātes plānošanas procesos un lietotņu izvietojšanas procesos;
 - c. sasniedzamais uzdevums: ietvars nodrošina publisko lietotņu izstrādei nepieciešamo funkcionalitāti, publiskās lietotnes, kuras nodrošina biznesa pamata funkcionalitāti.
6. Mašīnmācīšanās slāņa izveide:
 - a. aprobētais UIS-KDF ietvars lietojams MM slāņa izstrādes plānošanas procesā nosakot pieejas tiesības, funkcionalitātes, izvietojšanas un ātrdarbības prasības;
 - b. attīstības iterācijās ietvars lietojams jaunu modeļu izvietojšanas procesos;

- c. sasniedzamais uzdevums: zināšanu izguve no viedo sistēmu lietotāju reakcijas datiem, MM slānis, kurš nodrošina algoritmu izvietošanu un pielietošanu lēmumu pieņemšanas procesos.

7. Uzturēšana:

- a. uzturēšana ir iteratīva izstrādātās zināšanu izguves sistēmas pilnveidošana, un darbības nodrošināšana;
- b. aprobētais UIS-KDF ietvars sistēmas uzturēšanas procesā lietojams ieviešot izmaiņas un jaunus risinājumus, nodrošina zināšanu izguves sistēmas arhitektūras kopējo shēmu;
- c. sasniedzamais uzdevums: aprobētā UIS-KDF ietvara iteratīva papildināšana atbilstoši ieviestajām izmaiņām.

Izstrādājot zināšanu izguves sistēmas, UIS-KDF ietvars pielietojams veidojot un detalizējot sistēmas tehnoloģiju ietvaru. UIS-KDF ietvars aprobācijas procesā tiek koriģēts un pielietots atbilstoši noteiktās zināšanu izguves sistēmas prasībām rezultātā iegūstot sistēmas tehnoloģiju ietvaru. Tehnoloģiju ietvars veido sistēmas arhitektūras meta līmeņa skatu un izpratni par kopējo sistēmu.

3.3.1. PIELIETOŠANAS IEROBEŽOJUMI

UIS-KDF ietvara vispārējā arhitektūra, kurā ietilpst tehnoloģijas, publiskas un privātas lietojumprogrammas, mašīnmācīšanās un ietekmējošo kritēriju slāņi, ir daudzpusīga un piemērota lielai daļai nozaru, pateicoties tās pielāgojamībai un elastībai. UIS-KDF ietvars paredzēts zināšanu izguves sistēmu plānošanas agrai stadijai - infrastruktūras plānošanas atbalsta procesam par tehnoloģijām, to arhitektūru un plānošanu atbildīgajiem uzņēmumu darbiniekiem. UIS-KDF ietvara nolūks ir atvieglot zināšanu izguves sistēmu izstrādes sarežģīto un laikietilpīgo procesu, uzrādot arhitektūras plānošanas un organizācijas vadlīnijas.

UIS-KDF nav piemērots specializētām zinātniskās pētniecības jomām, piemēram, kvantu skaitļošanai. Kvantu skaitļošana ietver unikālas skaitļošanas pieejas un specializētus algoritmus, kas atšķiras no klasiskajām mašīnmācīšanās un datu analīzes metodēm. Kvantu skaitļošanas specifika nosaka prasības, kas būtu pielāgotas nozarei, var būt nepieciešamas atšķirīgas iekārtas, piemēram, kvantu procesori, un specializēta programmatūras vide, ko UIS-KDF neatbalsta.

Tiešsaistes tirdzniecības jomā finanšu tirgos UIS-KDF vispārējā arhitektūra var būt nepietiekama. Tiešsaistes tirdzniecībai nepieciešama augsta ātrdarbību un reāllaika datu

apstrādes iespējas, kam nepieciešamas optimizētas un specializētas pieejas. UIS-KDF daudzslāņu pieeja ar vairākiem abstrakcijas slāņiem (publiskās lietojumprogrammas, privātās lietojumprogrammas) var radīt aizkaves un sarežģītību, kas nav atbilstošas tiešsaistes tirdzniecībā. Nepieciešamība pēc tūlītējas datu apstrādes un ātras lēmumu pieņemšanas tiešsaistes tirdzniecībā izvirza nepieciešamību pēc optimizētas arhitektūras, kas īpaši izstrādāta, lai apstrādātu lielus datu apjomus ar minimālu aizkavi, ko vispārīgā sistēmai būtu sarežģīti sasniegt.

Eksistējošos lielos uzņēmumos un organizācijās (>250 nodarbināto) UIS-KDF var nebūt piemērots, jo šīs organizācijas bieži vien ir ārpus zināšanu izguves sistēmu projektēšanas un plānošanas agrās stadijas, kurai paredzēts UIS-KDF. Šādiem uzņēmumiem jau ir izveidotas, nobriedušas sistēmas un procesi, kas ir integrēti uzņēmumu esošajā IT infrastruktūrā un biznesa darbplūsmās. Lai ieviestu jaunu, vispārīgā pieeju, būtu nepieciešama apjomīga pārstrukturēšana. Pārejas izmaksas gan laika, gan resursu ziņā būtu apjomīgas. Šiem uzņēmumiem ir nepieciešami pielāgoti risinājumi, kas veidoti to īpašajām vajadzībām un darbības niansēm. Lielo uzņēmumu sarežģītība un inerce padara tos nepiemērotus UIS-KDF ietvara aprobācijai.

3.4. NODAĻAS KOPSAVILKUMS UN SECINĀJUMI

Pielietoto tehnoloģiju kopums un ietekmējošo faktoru (tiesisko prasību) apjoms padara zināšanu izguves sistēmu izstrādi par sarežģītu un laikietilpīgu procesu, kuru atvieglo zināšanu izguves ietvari, kuri nodrošina izstrādes vadlīnijas.

Ietvari paātrina izstrādes procesu, veicina atkārtotu izmantošanu un problēmu nošķiršanu, un nodrošina izstrādātājiem instrumentus elastīgāku un mazāk kļūdām pakļautu lietotņu izstrādei.

Neatbilstošu tehnoloģiju izvēle var novest pie nepietiekamas veiktspējas produkta izstrādes un potenciāliem sarežģījumiem piegādes un uzturēšanas procesos.

Neatbilstošas infrastruktūras izveide rada sarežģījumus gadījumos, kad zināšanu izguves sistēmai iteratīvi mainās prasības un esošā infrastruktūra neiekļauj jaunajām prasībām nepieciešamos resursus.

Mašīnmācīšanās izdalīšana sniedz iespēju atdalīt zināšanu izguves procesu (KDD procesu) no tehnoloģiju un lietojumprogrammu prasībām un funkcionalitātes.

Pieejamie zināšanu izguves ietvari nav tiešā veidā pārnesami starp dažādām nozarēm, autors izstrādājis UIS-KDF ietvaru, kuru iespējams aprobēt dažādās nozarēs.

UIS-KDF ietvars dalīts piecos loģiskajos blokos (slāņos) - publisko lietotņu, pārvaldes lietotņu, MM, ietekmējošo kritēriju un tehnoloģiju.

Zināšanu izguves sistēmu izstrādes procesu atvieglo zināšanu izguves ietvari, kas nodrošina izstrādes vadlīnijas. Zināšanu izguves ietvari tiek pielietoti zināšanu izguves sistēmu arhitektūras agrā izstrādes stadijā. Darba autora izstrādātais UIS-KDF ietvars pielietojams izstrādājot elastīgus un mērogojamus risinājumus zināšanu izguvei no viedo sistēmu lietotāju reakcijas datiem. UIS-KDF ietvara mērķis ir atvieglot zināšanu izguves sistēmu izstrādes procesu, nodrošinot arhitektūras plānošanas un organizācijas vadlīnijas.

Jansevskis & Osis (2024a) anonīmā grāmatas nodaļas recenzenta komentāri ar nozīmīgu pieredzi lietojumprogrammu izstrādē: *"There is good conceptual architecture of the proposed UIS-KDF. The definitions written are easy to read and the concepts are easy to understand. Based on the architecture definitions this is a conceptual document of a new system. If this is a concept document, there should be a freedom to choose technologies how to implement UIS-KDF, if all the requirements are met. For a software developer UIS-KDF might affect too much to the final implementation architecture when developing UIS-KDF systems."* Recenzenta ieteikumi iestrādāti UIS-KDF ietvarā.

Darba trešajā daļā:

- ir apkopoti un definēti zināšanu izguves sistēmu izstrādes posmi, izdalot piecus noteiktus posmus;
- ir izstrādāts un uzrādīts UIS-KDF zināšanu izguves ietvars, detalizējot ietvara slāņus;
- ir izstrādāta UIS-KDF ietvara pielietojuma kvantitatīvo rādītāju novērtējuma pieeja;
- ir izstrādāta UIS-KDF pielietojuma metodoloģija, kura apraksta, kā UIS-KDF ietvaru aprobēt organizācijā;
- ir identificēti UIS-KDF pielietošanas ierobežojumi.

Izstrādājot zināšanu izguves sistēmas, UIS-KDF ietvars lietojams veidojot un detalizējot sistēmas tehnoloģijas. UIS-KDF ietvars aprobācijas procesā tiek koriģēts un pielietots atbilstoši noteiktās zināšanu izguves sistēmas prasībām rezultātā iegūstot sistēmas tehnoloģiju ietvaru. Tehnoloģiju ietvars veido sistēmas arhitektūras meta līmeņa skatu un izpratni par kopējo sistēmu. UIS-KDF ietvara aprobācijas trīs veidi uzrādīti darba ceturtajā daļā.

4. UIS-KDF APROBĀCIJA

Šīs daļas ietvaros uzrādīti piedāvātā UIS-KDF ietvara aprobācijas trīs veidi:

- pirmais: teorētisks piedāvājums, apraksta ietvara aprobāciju aktīvās mācīšanās atbalstam, izstrādāts Eiropas Sociālā fonda darbības programmas "Izaugsme un nodarbinātība" 8.2.2. specifiskā atbalsta mērķa "Stiprināt augstākās izglītības institūciju akadēmisko personālu stratēģiskās specializācijas jomās" projekta "Vidzemes Augstskolas akadēmiskā personāla pilnveide un cilvēkresursu attīstība" (Nr.8.2.2.0/18/A/012) ietvaros;
- otrais: ietvara ieviešana jaunuzņēmuma Monetization Solutions Inc. (*Monetization Solutions Inc*, 2022) un tā meitas uzņēmuma Latvijā, SIA Monetization Solutions, Reģistrācijas nr. 40203021882, darbības nodrošināšanai un pētniecības projektu realizācijai;
- trešais: uzrāda UIS-KDF ietvara mašīnmācīšanās slāņa viena algoritma apmācības detalizētu izstrādi, projekts realizēts uzņēmuma SIA Monetization Solutions pētniecības projekta ietvaros. Darba autors ir minētā jaunuzņēmuma pētniecības projektu autors, autoram ir tiesības pielietot pētījumus promocijas darba izstrādē.

4.1. AKTĪVĀS MĀCĪŠANĀS ATBALSTS

Tradicionālās lekcijas, kurās autors pielieto iepriekš sagatavotus materiālus ar ierobežotām mijiedarbības iespējām, nav efektīvākais veids un lielākoties nenodrošina kvalitatīvu mācību procesu. Aktīvā mācīšanās ir pieeja, kas ietver studentu aktīvu iesaistīšanu kursa procesā, izmantojot diskusijas, problēmu risināšanu, gadījumu izpēti, lomu spēles un citas metodes. Uz studentiem vērstās mācīšanas pieejas uzliek studentam lielāku atbildību nekā pasīvās pieejas, piemēram, tradicionālās lekcijas. Aktīvās mācīšanās un sadarbības aktivitātes var ilgt no dažām minūtēm līdz vairāku stundu sesijām un var notikt vairāku mijiedarbību laikā (Jansevskis & Osis, 2020a).

Zināšanu izguvi aktīvajā mācību ekosistēmā var veidot, uzkrājot un analizējot datus no konkrētās mācību vides. Apkopotos lietotāju reakcijas datus var izmantot, lai labāk izprastu auditoriju, gūtu ieskatu un veicinātu aktīvu atgriezenisko saiti ar studentiem.

Aktīvā mācīšanās (AM) ir pieeja, kas veicina mācīšanu, pielietojot uz studentiem orientētas metodes, un atbalsta mācību priekšmeta zināšanu un izpratnes attīstību, patstāvīgu mācīšanos un digitālo izpratni. Viena no AM metodēm ir Projektu Bāzēta

Mācību metode (PBM), kurā studenti nodarbību laikā parasti tiek sadalīti grupās no diviem līdz pieciem dalībniekiem. Piemēri ar PBM kā aktīvas mācīšanās pieeju uzrāda, ka studenti uzskata, ka viņu mācīšanās un līdzdalība AM nodarbībās ir pozitīva (Ferreira & Canedo, 2020).

Wiggins et.al (Wiggins et al., 2017) definē aktīvās mācīšanās klases (AMK) kā sociālu aktivitāti, kas motivē auditorijai mijiedarboties un mācīties no saviem kolēģiem. Pētījumā Wheeler (Wheeler, 2018) apraksta, kā mācību vides plānojums, izkārtojums un telpas iekārtojums spēj veicināt indivīdu sadarbību un problēmu risināšanas spējas, kas virza auditoriju uz orientētu mācīšanos. Zināšanu izguves ietvari AMK palīdz iegūt holistiskāku skatījumu uz auditorijas pieredzi (Jansevskis & Osis, 2020a).

UIS-KDF ietvara aprobācijas pieeja veidota konceptuāla, piemērojot mašīnmācīšanās risinājumus aktīvās mācīšanās atbalstam kā daļu no lielāka meta-ietvara. Aprobācijas procesā darba autors pārveidoja docētos kursus atbilstoši aktīvās mācību klases pieejai, izpildot četrus secīgus uzdevumus:

1. Sagatavošanās pārejai uz AMK;
2. Kursa daļēja pārveidošana par AMK;
3. Visu materiālu secīga pārveidošana AMK pieejai;
4. Integrācijai pieejamo metriku uzkrāšana, uzlabošana un analīze.

Minētie soļi pielāgoti un papildināti no Drake (Drake, 2015) trīsdesmit četru sagatavošanas apakšsoļu saraksta. Sākotnējā posmā docētājiem jāiepazīstas ar tehnoloģijām, pieejamo programmatūru un aparatūru auditorijā, kā arī jāapgūst aktīva sadarbība ar studentiem. Otrajā posmā docētājiem jāpārveido kursa materiāli aktīvai mācīšanās pieejai, izmantojot vienu vai vairākas uz studentiem orientētas mācību metodes.

Autors savos docētajos studiju kursus īstenoja uz projektiem balstītu mācīšanos, koncentrējoties uz praktiskām pieejām un mācīšanos caur pieredzi. Trešajā posmā jāiegūst un jāapkopo rādītāji un atgriezeniskās saites dati, lai novērtētu izvēlēto aktīvās mācīšanās pieeju. Ceturtais solis ievieš nepārtrauktas izstrādes un ieviešanas stratēģiju un iteratīvos ciklus ieviešanas pieejai. Ieviestā aktīvās mācību klases integrācijas metodika atspoguļo veiklu programmatūras izstrādi ar pielietojumu izglītībā informācijas tehnoloģiju bakalaura studiju programmas ietvaros.

Pētījums (Jansevskis & Osis, 2020a) ietver potenciālo datu kopu noteikšanu, kuras iespējams uzkrāt, lai izstrādātu mašīnmācīšanās algoritmus, kas atbalstītu aktīvu mācību ekosistēmu. Potenciāli pieejamā datu kopa definē nozīmīgākās kategorijas, kuras iespējams uzkrāt turpmākai algoritmu izstrādei (skat., 4.1.tabulu).

Pastāv vairākas mašīnmācīšanās lietotnes, kas paredzētas izglītības sfērai, tādās jomās kā satura analīze, plānošanas algoritmi, vērtēšanas sistēmas, mācību analītika un kognitīvā psiholoģija. IBM Watson Explorer Content Analytics var apkopot un analizēt strukturētu un nestrukturētu saturu dokumentos, e-pastos, datu bāzēs, vietnēs un citās datu krātuvēs, piemēram, e-mācību sistēmu datos (*IBM Watson Explorer, 2022*). Adaptīvā mācību tehnoloģija spēj pielāgoties studentu zināšanu līmenim katrā mijiedarbībā, lai noskaidrotu kādas papildus darbības tiem būtu jāveic, lai uzlabotu sniegumu - atgriezeniskā saite tiek nodrošināta laikā, kad studenti izpilda uzdevumus (Jansevskis & Osis, 2020a).

4.1. tabula. Lietotāju reakcijas datu potenciālās kopas (adaptēts no Jansevskis & Osis, 2020)

Kategorija	Apakškopa	Paraugs
Snieguma dati	Studenta sniegums kursa ietvaros	Kursa gala atzīme
Mācību procesa atbalsta sistēmas dati	Dati no mācību procesa atbalsta sistēmas	Dokumenti, kursu materiālu saturs, individuālie uzdevumu vērtējumi, mijiedarbības dati
Demogrāfijas dati	Bināri dati	Dzimums, vecums
Informācija no cilvēkresursu sistēmas	Iepriekšējās zināšanas un profesionālā pieredze	Vidusskola, apgūtie papildus kursi, iepriekšēja izglītība
Reģistrācijas informācija	Iepriekšēja reģistrācija, sākotnējā darbība	Laika zīmogu dati
Programmatūras lietotāju reakcijas dati	Dati, iegūti no aparatūras un iekārtām, kas pieejamas mācību vidē	Mitruma, temperatūra, laiks, citi

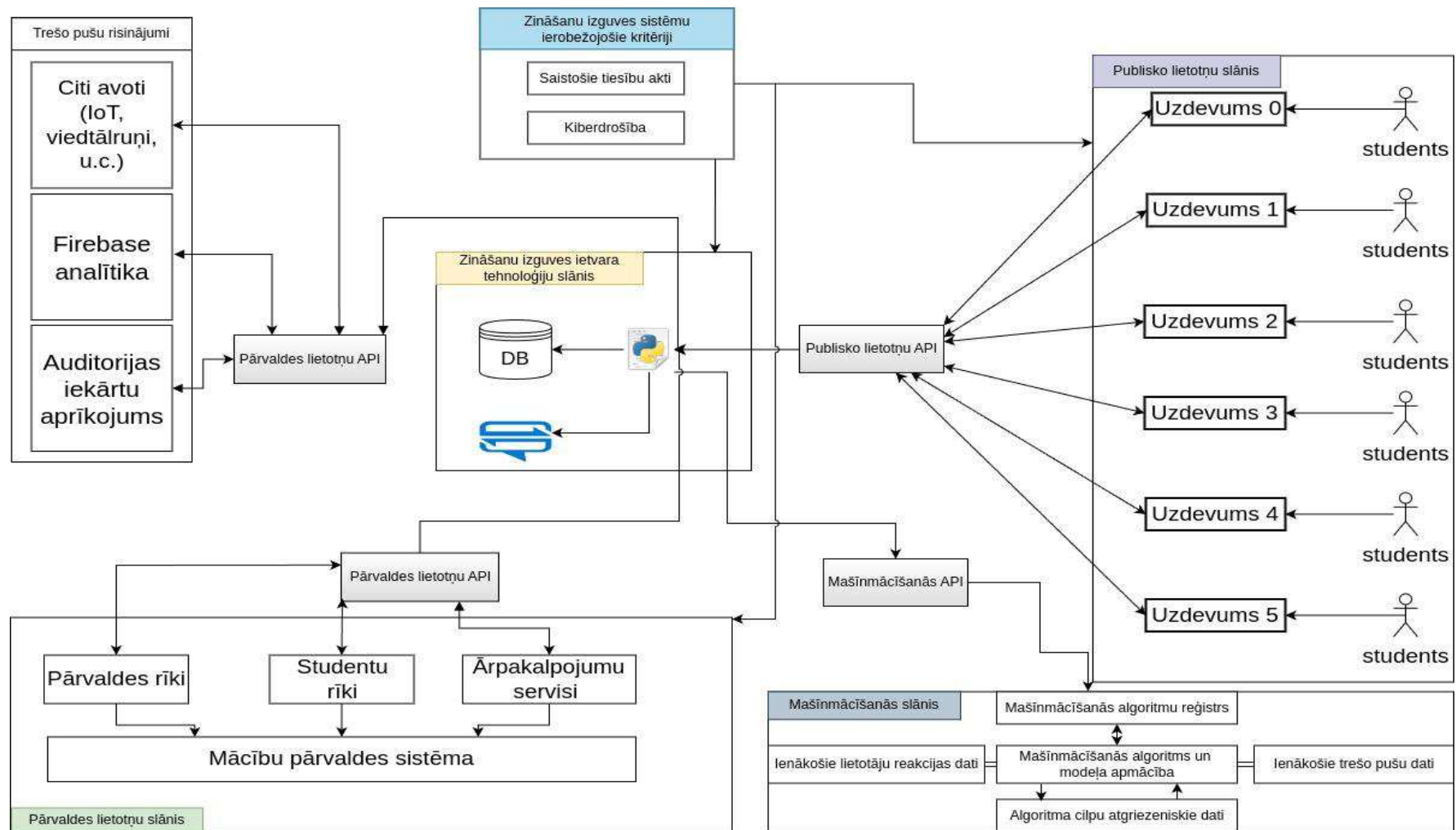
Aktīvās mācību klases ekosistēma ietver ne tikai programmatūras risinājumus, bet arī aparatūras aprīkojumu, kas integrēts mācību vidē. Aktīvā mācību ekosistēmā ietilpst auditorijas skārienekrāna monitori, skaņas sistēmas, vadības procesori, vadības monitori, skārienjutīgie sensori, mobilās iekārtas, kameras, sensori (temperatūras, mitruma, ogļskābās gāzes), kā arī dati no Wi-Fi un lokālajiem tīkliem. Aktīvās mācību klases ekosistēma nodrošina datus, kurus var pielietot zināšanu izguves sistēmas izveidei.

Pētījumā (Jansevskis & Osis, 2020a) paplašināta pieejamo datu kopa ar papildu datiem, kurus var iegūt, veidojot praktiskus uzdevumus programmatūras izstrādesursos. Piemēram, praktiskais programmatūras izstrādes uzdevums, kurš ietver vienkāršas ziņu plūsmas mobilās lietotnes izstrādi un reģistrē katru sasniegto uzdevuma stāvokli.

Hattie un Timperley (Hattie & Timperley, 2007) atgriezenisko saiti konceptualizē kā aģenta sniegto informācija – aģents var būt pasniedzējs, kas sniedz informāciju, vai arī datorsistēma. Lietotāju reakcijas dati ir informācija, kas saistīta ar uzdevumu vai mācīšanās procesu, kas aizpilda plaisu starp to, kas tiek apgūts, un to, ko plānots apgūt (Sadler, 1989). Hattie un Timperley (Hattie & Timperley, 2007) ierosina, ka atsauksmēm augstākajā izglītības sistēmā jāatbilst trim nosacījumiem: mērķu definēšana, mērķu sasniegšana, virzība uz mērķa sasniegšanu un jautāšana, kādas darbības jāveic, lai studenti pilnveidotu zināšanas. Pētījumā (Jansevskis & Osis, 2020a) ierosināts tehnoloģiju ietvars zināšanu izguvei no viedo sistēmu lietotāju reakcijas datiem aktīvās mācīšanās atbalstam (skat. 4.1.att.), kas balstīts uz izstrādāto UIS-KDF ietvaru (skat. 3.1.att.).

Aktīvās mācīšanās ietvars zināšanu izguvei no viedo sistēmu lietotāju reakcijas datiem (skat 4.1.att.) veidots kā komplekss nodrošinot paplašināšanas iespējas. Tehnoloģiju ietvara būtiskākās komponentes saistītas ar darba trešajā daļā izstrādāto UIS-KDF ietvaru (skat 3.5. att.). Zināšanu izguves ietvars veidots no vairākām API vārtējām atšķirīgiem nolūkiem — lietotāju reakcijas datu uzkrāšanai, e-mācību platformām, mācību vides iekārtu savienošanai, mobilajām lietotnēm un analītikas lietotnēm.

Saskaņā ar Rietsche et al. (Rietsche et al., 2018), aktīvajai lietotāju reakcijas datu sistēmai jābūt tīmekļa lietotnei ar lietotājiem draudzīgu saskarni. Šai prasībai pētnieki Rietsche et. al. piedāvā modernu lietotāju saskarņu izstrādes platformu, piemēram, Angular, React vai Vue. Turklāt pētnieki definē specifiskas prasības – zināšanu izguves no lietotāju reakcijas datiem sistēmai jābūt viegli integrējamai esošajās universitāšu e-studiju infrastruktūrās un autentifikācijas sistēmās.



4.1.attēls. Adaptēts UIS-KDF ietvars zināšanu izguvei no viedo sistēmu lietotāju pieredzes datiem aktīvās mācīšanās atbalstam (fragments, adaptēts no Jansevskis & Osis, 2020).

Studentiem jāspēj individuāli novērtēt sevi un vienlaikus salīdzināt savu sniegumu ar vidējo rādītāju. Lai zināšanu izguves sistēma nodrošinātu ilgtermiņa lietderīgumu, tai jābūt elastīgai un paplašināmai, lai ieinteresētās puses varētu izmantot pieejamās datu kopas un izvietot papildu rīkus kā arī pārbaudīt hipotēzes. Lai izpildītu šo nosacījumu, sistēmai ieviestas komunikācijas vārtejas pielietojot API (skat. 4.1.att.). Šāda pieeja nodrošina paplašināšanas un savietošanas iespējas starp dažādām eksistējošām universitāšu sistēmām. Pielietojot API vārtejas kā galvenos piekļuves punktus, infrastruktūru pielieto, lai izstrādātu specifiskus un praktiskus uzdevumus programmatūras izstrādes kursiem, piemēram, individuālu progresa izsekotāju vai iekštelpu klimata uzraudzības lietotņu izstrādei.

UIS-KDF ietvara aprobācija aktīvās mācīšanās atbalstam izstrādāta teorētiskā līmenī, praktiskā lietošanā ieviešot autora docētajos studijuursos projektu bāzētu aktīvās mācīšanās metodi un praktisko darbu izstrādi. Lietotāju reakcijas datu nodrošināšanai pielietotas augstskolas eksistējošās e-mācību platformas iespējas. Zināšanu izguves ietvara pielietošana aktīvās mācīšanās atbalstam uzskatāma par aprobācijas pieeju, kurai nepieciešama papildus izstrāde. Savukārt otrais zināšanu izguves ietvara aprobācijas piemērs apskata UIS-KDF ietvara pielietojumu uzņēmējdarbības mērķu atbalstam un iekļauj arī praktisku pielietojumu.

4.2. MOBILO LIETOTŅU NOZARE

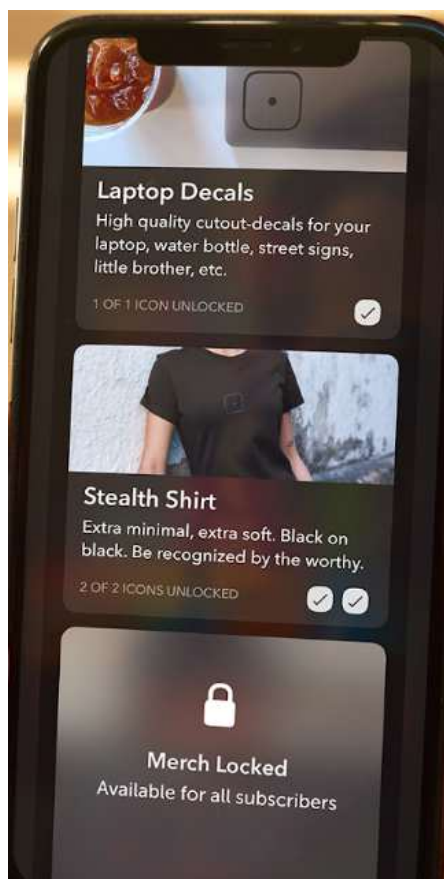
Spēļu nozare ir strauji augoša ar lielu globālo tirgu, un tiek prognozēts, ka līdz 2025. gadam tā sasniegs 300 miljardus USD. Zināšanu izguve tiek panākta uzkrājot un analizējot datus no spēļu lietotnēm. Apkopotie dati tiek izmantoti, lai izprastu spēlētājus, gūtu ieskatu un uzlabotu produktus spēļu kopienai, pamatojoties uz atsauksmēm un lietotāju mijiedarbību. Rezultātā iegūtās zināšanas veicina monetizāciju tādā veidā, kas ir saistošs e-sporta un straumēšanas telpā (Jansevskis & Osis, 2020).

Mobilajās spēlēs parasti pastāv nevienlīdzīgs dalījums starp spēlētājiem, kuri ir maksājoši, un tiem, kuri tādi nav. Viens no izaicinājumiem mobilajās spēlēs ir pārveidot spēlētājus no nemaksājošajiem uz maksājošajiem, tādējādi nodrošinot, ka spēlētāju vērtība (angliski: Life Time Value - LTV) ir augstāka par spēlētāju ieguves izmaksām (angliski: User Acquisition Cost - UAC) (Sifa et al., 2015). Lai palielinātu LTV un samazinātu UAC, zināšanu izguve, uzkrājot un analizējot datus no spēlēm tādā veidā, kas atbilstu VDAR, jāīsteno tā, lai tas negatīvi neietekmētu spēlētājus. Darba autori pētījumā (Jansevskis &

Osis, 2020) par mērķi izvirza noskaidrot, vai esošie kādas spēles lietotāji vai ierīces var kļūt par maksājošiem spēlētājiem.

Pētījumam (Jansevskis & Osis, 2020) definēti trīs uzdevumi – noteikt potenciālo datu kopu, kuru var uzkrāt, ievērojot VDAR nosacījumus, izstrādāt mobilās pakotnes integrēšanai noteiktās mobilajās spēlēs, uzkrāt datus analīzes veikšanai un pielietot mašīnmācīšanās algoritmus. Pētījumā izstrādātais tehnoloģiju ietvars balstīts uz iepriekšējiem pētījumiem zināšanu izguves jomā, izmantojot mašīnmācīšanos (Jansevskis & Osis, 2018), kā arī pielietots izstrādātais UIS-KDF ietvars (skat., 3.5.att.), kurš ieviests praktiskā izmantošanā uzņēmumā Monetization Solutions Inc (Monetization Solutions Inc., 2022). Lai pētījumu savienotu ar uzņēmējdarbības mērķiem, promocijas darba autors izstrādāja tehnoloģiju ietvaru, kurš uzkrāj datus, piedāvā mobilās pakotnes integrēšanai eksistējošās mobilo spēļu lietotnēs (skat., II, III, IV, V pielikumi), kā arī izveidoja mašīnmācīšanās risinājumu (skat., VI pielikums).

Pētījuma procesa laikā definēta potenciālā datu kopa, kuru var pielietot MM algoritmu apmācībai. Pieņēmumi balstīti uz līdzīgiem pētījumiem Sifa et.al. (2015), Han & Windsor (2013), Pantano & Priporas (Pantano & Priporas, 2016) un Xudong (2016) un potenciāli pieejamajiem datiem, kurus var iegūt no spēlēm. Dati apkopoti par spēlētāja uzvedību sekojošās kategorijās: sesijas, produktu skatījumi un pirkumi. Sesija tiek reģistrēta, kad spēlētājs uzsāk spēli ar iekļautu izstrādāto pakotni. Produktu skatījums tiek reģistrēts, kad spēlētāja mobilā iekārta pieprasa produktu datus no servera API. Pirkums tiek reģistrēts, kad spēlētājs iztērē līdzekļus par spēles produktiem un noformē pasūtījumu. Lai padarītu šo pētījumu atkārtojamu, izvēlēts savienot mobilās pakotnes ar tiešsaistes veikalu un nodrošināt produktus iegādei. Tiešsaistes veikalam pievienoti produkti - t-krekli, zeķes, dāvanu kartes un 3D figūriņas ar pielāgotu dizainu katrai spēlei (skat., 4.2.att.).



4.2.attēls. Mobilās pakotnes produktu skats, pielāgots noteiktai spēlei (autora veidots).

Mobilo pakotņu uzkrājamajai datu kopai (skat., 4.2.tabulu) kā prasība noteikta atbilstība VDAR un tajā netiek iekļauta personu informācija (skat., VII pielikumu). Tā kā lietošanas procesā tiek veikti pirkumi un nepieciešams pasūtījumu izpildes process, ir nepieciešams iegūt personas datus piegādes nodrošināšanai. Lai izpildītu VDAR atbilstību, pielietota Shopify tiešsaistes platforma, kas nodrošina personas datu glabāšanu, tādējādi dati netiek glabāti uzņēmuma datubāzēs un mērķa datu kopa neiekļauj spēlētāju personas informāciju.

4.2. tabula. Pētījumam noteiktā lietotāju reakcijas datu kopa (autora veidota)

Kategorija	Veids	Paraugs
Iekārtas dati	Iekārta OS versija Valoda Apple/Google Pay statuss	iPhone10.6 9,1 en_US Aktīvs
Spēles dati	Kategorija Izdevējs	Puzles Blizzard

Kategorija	Veids	Paraugs
	Kopējais spēlētāju skaits Uzstādīšanas datums Atjaunināšanas datums Produkts	1224 2020-02-20 2020-02-20 t-krekls
Lietotāja	Spēles laiks Pēdējā sesija Sesijas ilgums Sesijas sākums Sesijas beigas	455 2020-02-20 707 2020-02-20 20:20:06.56554+02
Mijiedarbības	Aizvēršanas darbības Aizvēršanas darbību skaits Aizvēršanas koeficients	3 45 1
Pirkšanas uzvedība	Tēriņa summa Pirkumu skaits Valūta	6,52 1 USD
Klienta	Pilsēta Novads Valsts Produkts	Atlanta Georgia US t-krekls

Algoritma apmācības procesa priekšapstrādē iekļauta datu attīrīšana, kategoriju mainīgo definēšana un mainīgo normalizēšana. Priekšapstrādes uzdevums ir izņemt ierakstus ar trūkstošiem datiem, izņemt ierakstus ar konstantām vērtībām un izņemt ierakstus, kurus var noteikt tikai pēc pirkumu izpildes. Procesā izpildes laikā bija zināms, ka dati iekļaus pārsvarā nemaksājošus spēlētājus, jo tikai neliela daļa visu mobilo lietotņu spēlētāju ir maksājoši klienti (Sifa et al., 2015). Lai risinātu nelīdzsvarotu datu kopu problēmu, testēšanas nolūkos ģenerēti maksājoši spēlētāji, izmantojot SMOTE-NC metodi (angliski: Synthetic Minority Over-sampling Technique-Nominal Continuous) (Chawla et al., 2002).

Lai novērtētu F-score vērtību (angliski: F1-Measure) (Wood, 2020), izmantots vidējais harmoniskais rādītājs, lai prognozētu maksājošos spēlētājus. Izmantots arī katras klases klasifikācijas precizitātes vērtību ģeometriskais vidējais (angliski: G-mean) (Sarpong, 2017). Bāzes risinājumam apmācībai izvēlēti sekojoši modeļi:

- lēmumu koki (angliski: decision trees with reduced error pruning);
- atbalsta vektori (angliski: support vector machines with polynomial and radial basis function kernels);

- nejauši meži (angliski: random forests).

Uzmanība pētījumā pievērsta to faktoru noteikšanai, kuriem ir būtiska ietekme uz prognozi vai spēlētājs būs maksājošs klients. Pastāv divas metodes, kas iesaka produktus klientam - pirmā balstīta uz lietotāja reakcijas vēsturiskajiem datiem un otrā uz saturu produktu aprakstos un to, kā tie atbilst spēlētāja profilam. Pirmā risinājuma priekšrocība ir detalizēti aprakstīta kopā ar produktu atlasītiem klientiem Candillier et al. grāmatā (Candillier et al., 2009), būtiskais trūkums minētajā metodē ir tas, ka spēlētājam nepieciešami vēsturiskie dati.

Sadarbības filtrēšanai (angliski: collaborative filtering) ir augstāka precizitāte, un tā tiek izstrādāta, pamatojoties uz dziļās mācīšanās metodēm (L.-S. Chen et al., 2017). Šī pieeja saskaras ar izaicinājumu - mērogojamības problēmu. Darbā Wei et al. (Wei et al., 2017) tiek piedāvāta aukstā starta (spēlētāji bez vēsturiskiem datiem) izaicinājuma risināšanas pieeja. Pamatojoties uz minētajiem pētījumiem, bāzes risinājums izveidots balstoties uz vienumu-vienumu sadarbības filtrēšanu.

Plānojot pirkuma uzvedības analīzes ietvaru, noteikts pielietojamo tehnoloģiju kopums, kas aptver mobilajās spēlēs biežāk izmantotās tehnoloģijas. 4.3. attēlā attēlota ietvara struktūra, lai īstenotu spēlētāju iepirkšanās uzvedības analīzi. Izstrādātais tehnoloģiju ietvars balstīts uz izveidoto UIS-KDF ietvaru, detalizējot tehnoloģisko steku, lai atbalstītu uzņēmuma biznesa mērķus. Izstrādātais tehnoloģiju ietvars sastāv no pieciem slāņiem, atbilstoši UIS-KDF ietvara slāņiem, kuri nodrošina datu uzkrāšanu, tas izstrādāts ap centrālo datu glabātuvī, kurai var piekļūt, izmantojot lietojumprogrammu saskarnes.

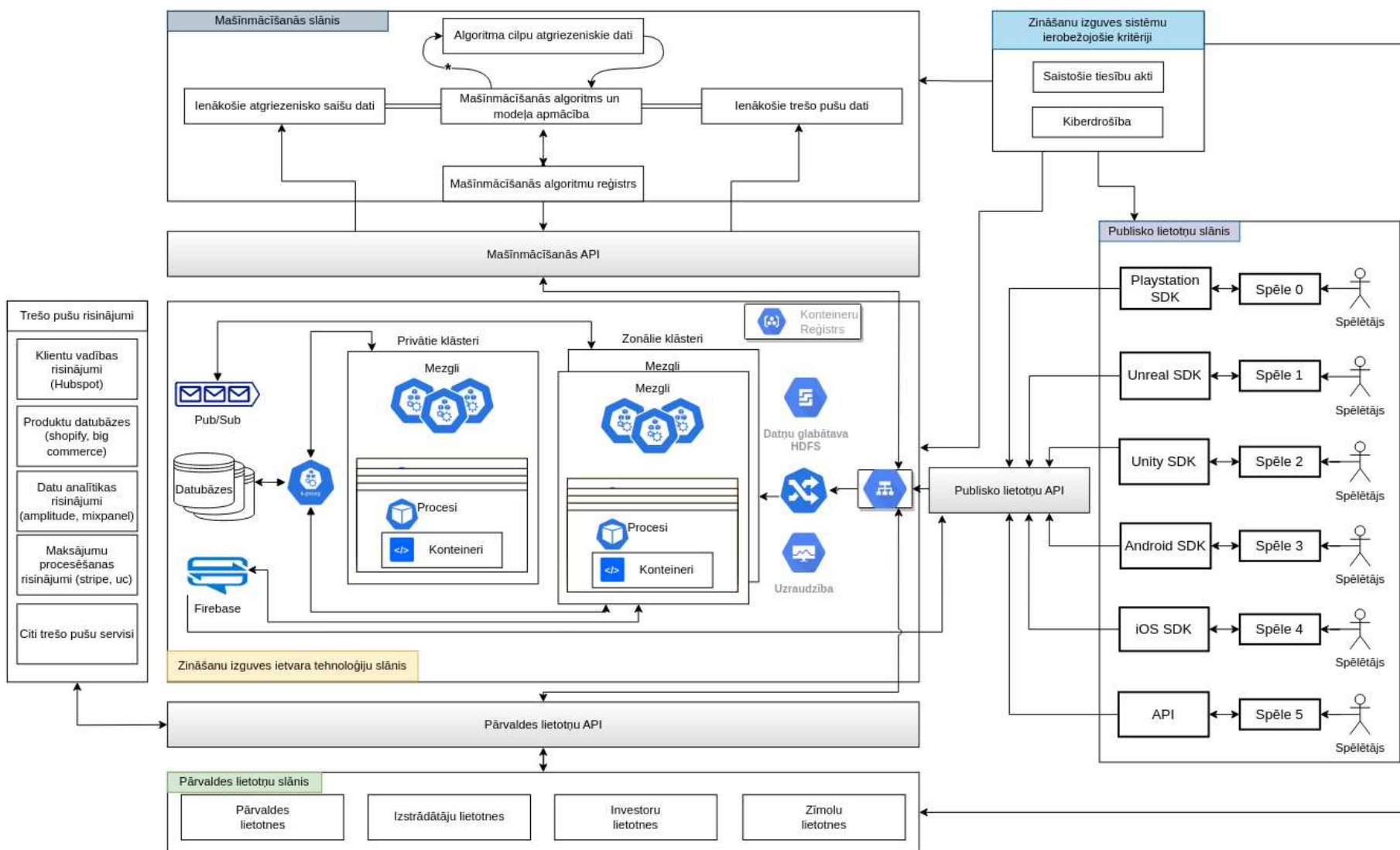
Datu glabāšana izveidotajā tehnoloģiju ietvarā nodrošina tādu funkcionalitāti kā datu nepārtrauktība, integritāte un pieejamība. Mākoņu krātuves viens no uzdevumiem ir nodrošināt datus mobilajām pakotnēm. Tā kā mobilo spēļu un spēlētāju iekārtu radītais datu apjoms ir ievērojams, būtisks izaicinājums ir nodrošināt risinājumus, kuri ir paplašināmi un mērogojami.

Drošības apsvērumi un uz lomu balstītas piekļuves kontroles (angliski: Role Based Access Controls - RBAC) ieviestas starp visām tehnoloģiju ietvara komponentēm. Lai nodrošinātu lietotāju privilēģiju tehniskās prasības, pētījumā pielietots OAuth 2.0 (Parecki, 2022) nozares standarta protokols katrai mobilās programmatūras izstrādes pakotnei un ierobežota piekļuve tikai lasīšanas režīmā publiski pieejamajai lietojumprogrammu saskarnei. Lai noteiktu drošības apsvērumus, pielietots drošības brieduma modelis BSIMM (angliski: Building Security in Maturity Model) (Synsopsys, 2022).

Lai uzkrātu lietotāju reakcijas datus no mobilo spēļu lietotnēm, mobilās pakotnes visbiežāk sastopamajām platformām izstrādātas un savienotas ar publisko lietojumprogrammu saskarni (V pielikums). Tehnoloģiju ietvara izstrādes procesā izveidotas mobilās pakotnes Android (II pielikums), iOS (III pielikums) un Unity (IV pielikums) platformām. Lietotāju reakcijas dati uzkrāti arī no lietotnēm, kurās pielietotas maksājumu platformas Apple Pay (*Apple Pay*, 2022), Google Pay (*Google Pay*, 2022) un Stripe (*Stripe*, 2022).

Lai pārbaudītu pētījuma procesā apmācīto algoritmu, sagatavota lietotāju reakcijas datu kopa, balstoties uz sākotnēji noteikto potenciālo datu kopu (skat., 4.2.tabulu). Par reakcijas datu primāro atslēgu izvirzīts iekārtas identifikators (šifrēts lietojot vienvirziena kriptogrāfijas algoritmus pirms ievietošanas datubāzē), kas identificē unikālu ierīci. Balstoties uz ierīces identifikatoru, apkopoti dati par pirkumiem tiešsaistes veikalos un izmantoti apmācību uzdevumam. Priekšapstrādes procesā sagatavota lietotāju reakcijas datu apakškopa, izmantojot datus no divu nedēļu laika posma ar vairāk nekā 50 tūkstošiem ierakstu, kas satur aptuveni 500 monetizētas iekārtas. Apmācāmie modeļi pārbaudīti pielietojot līdzsvarošanu, jo tikai aptuveni 1% ierīču monetizētas.

Pielietotas divas līdzsvarošanas pieejas - SMOTE-NC un Random OverSampling. Izmantoti sekojoši klasifikācijas modeļi: Dummy klasifikators, AdaBoost klasifikators, Logistic Regression un Random Forest klasifikatori. AdaBoost klasifikatora modelis ar Random OverSampling līdzsvarošanas metodi nodrošināja daudzsološākos rezultātus ar precizitāti 0,854158. Šim modelim atgriezti 85,03% patiesi negatīvu un 0,39% patiesi pozitīvu rezultātu, kas norāda, ka aptuveni 50% monetizēto ierīču atpazītas korekti, savukārt tikai neliela daļa no monetizētajām ierīcēm atpazītas nekorekti.



4.3.attēls. Adaptētais UIS-KDF ietvars (adaptēts no Jansevskis & Osis, 2020).

Tabulā (skat., 4.3.tabula) sniegts kvantitatīvo mērījumu apkopojums katram UIS-KDF slānim aprobācijas scenārijā uzņēmumā Monetization Solutions Inc. Pielietotas ietvara kvantitatīvai novērtēšanai izvirzītās pazīmes, kuras apkopotas 3.2.tabulā, pielāgojot specifiskajam aprobācijas scenārijam. Šie rādītāji palīdz uzņēmuma lēmumu pieņēmējiem un ieinteresētajām personām novērtēt zināšanu izguves sistēmas darbību un pieņemt lēmumus par tās aprobāciju un pilnveidošanu.

4.3. tabula. UIS-KDF aprobācijas izvērtējuma apkopojums (autora apkopota, Monetization Solutions Inc. dati)

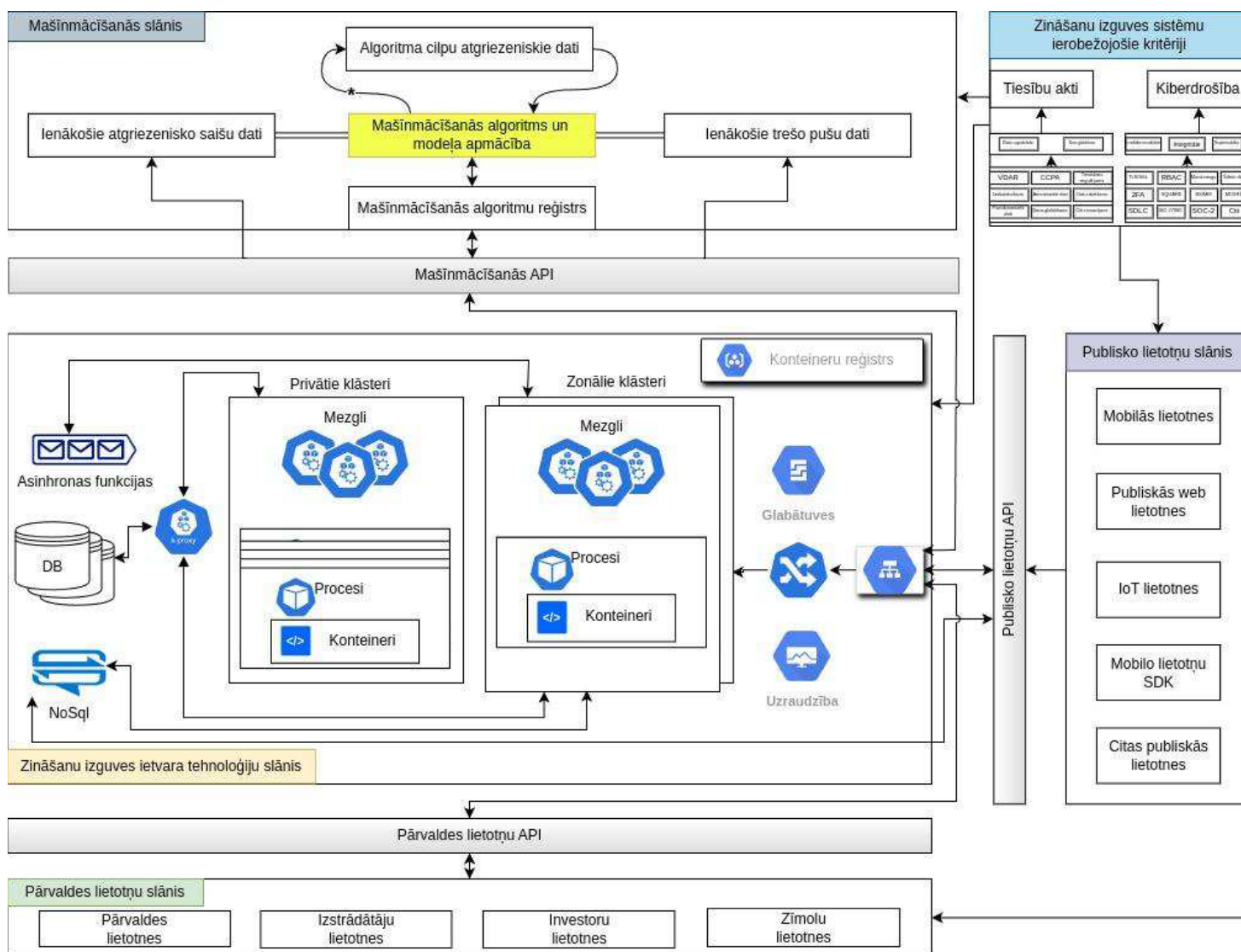
Slānis	Metrika	Mērījums
Tehnoloģiju slānis	Datu integrācijas ātrums	8 stundas, viena darba diena.
	Darbspējas laiks	99.98% gada griezumā.
	Datu apstrādes ātrums	273ms - 970ms atkarībā no klienta fiziskās atrašanās vietas.
Publisko lietotņu slānis	Lietotāju pieņemšanas līmenis	30 miljoni unikālu lietotāju 2024.gada aprīlis.
	Lietotāju apmierinātības līmenis	Nav apkopots
	Sistēmas atbildes ātrums	273ms - 970ms atkarībā no klienta fiziskās atrašanās vietas.
Privāto lietotņu slānis	Datu drošības incidenti	Trīs zemas nozīmes incidenti divu gadu laikā
	Piekļuves laiks	1500 - 3000 milisekundes
	Atbilstības standartiem līmenis	89% VDAR (2024.gada maijs) 95% SOC-2 (2024.gada maijs)
Mašīnmācīšanās slānis	Modeļu precizitāte	85%, apmācību nepieciešams atkārtot pielietojot lielāku datu apjomu
	Apmācības laiks	Nav apkopots
	Prognozes latentums	300ms apmācītajam algoritmam uzstādītam un lietotam kā klāstera konteinerim
Ietekmējošo kritēriju slānis	Lēmumu pieņemšanas uzlabojums	Nav apkopots

Slānis	Metrika	Mērījums
	Peļņa no kapitāla ieguldījuma	7%
	Uzņēmējdarbības ietekmes vērtējums	Nav apkopots

Pētījuma procesā izstrādātais UIS-KDF ietvars (skat. 3.5.att.) apobēts izmantošanai mobilo lietotņu nozarē, izveidojot tehnoloģiju ietvaru un infrastruktūru, un mobilās pakotnes kā arī pārbaudot tehnoloģiju ietvara darbību divās publiski pieejamās mobilajās lietotnēs - Blackbox (*Blackbox*, 2022) un Slopes (*Slopes*, 2022). Pētījuma procesā apmācītais algoritms, kura mērķis ir prognozēt vai noteikta iekārta var tikt monetizēta, ir konceptuāls modelis, kuru praktiskai pielietošanai nepieciešams atkārtoti testēt un apmācīt pielietojot lielāku datu kopu. UIS-KDF aprobācija jaunuzņēmuma darbības nodrošināšanai vienkāršoja uzņēmuma infrastruktūras izstrādes procesu, modulārais dalījums slāņos spēj nodrošināt atbilstošu globālu mērogojamību uzņēmumam paplašinot un diversificējot darbību. UIS-KDF pielietošana sniedza iespēju veikli sagatavot uzņēmumu SOC-2 sertifikācijas procesam un iekšējo procesu sagatavošanā VDAR atbilstībai. UIS-KDF modularitāte nodrošina mazāko tiesību principu (angliski: principle of least privilege), kā arī izstrādātājiem sniedz iespējas lietot tiem zināmās tehnoloģijas abstrahējoties no kopējās infrastruktūras sarežģītības (Monetization Solutions Inc., 2022). Trešais aprobācijas piemērs apraksta izstrādātā UIS-KDF ietvara mašīnmācīšanās komponentes moduļa viena algoritma pielietojuma detalizētu izstrādi.

4.3. MOBILO IEKĀRTU BATERIJU SLODZES PROGNOZES

Mobilo iekārtu, mobilo tālruņu un mobilo viedtālruņu straujā izplatība padara mobilās tehnoloģijas par nozīmīgu datu uzkrāšanas instrumentu (Barzilay, 2019). Aprobācijas mērķis ir prototipa algoritma apmācība, kura uzdevums ir prognozēt iekārtas akumulatora enerģijas patēriņu dažādos lietošanas scenārijos. Veiktā pētījuma nolūks ir spēt noteikt vai uzņēmuma Monetization Solutions Inc. izstrādātās mobilo lietotņu pakotnes var ietekmēt mobilās iekārtas negatīvā veidā, tā, ka to varētu ievērot mobilās iekārtas gala lietotājs. Aprobācija demonstrē UIS-KDF ietvara mašīnmācīšanās slāņa viena algoritma apmācības procesu (skat., 4.4.att.).



4.4.attēls. UIS-KDF ietvara mašīnmācīšanās slāņa viena algoritma apmācības process (autora veidots, izgaismots atbilstošais elements).

Pieejams salīdzinoši liels skaits pētnieciskās literatūras par mobilo iekārtu lietojumu izmantošanas ietekmi uz iekārtas akumulatora stāvokli, piemēram, Metri et al. (Metri et al., 2012), Pramanik et al. (Pramanik et al., 2019), Carrol un Heisner (Carrol & Heisner, 2010) publikācijas, kurās pētnieki sniedz detalizētu analīzi par viedtālruņa akumulatora enerģijas patēriņu saistībā ar lietošanas metodēm, vienlaikus piedāvājot optimizācijas metodes. Pētnieciskā literatūrā tomēr neiekļauj iekārtas, kuras ir relatīvi jaunas un spējīgas izpildīt mašīnmācīšanās algoritmus.

Pētījuma pirmajā posmā izstrādāta datu uzkrāšanas metodika un testēšanas scenāriji, lai noskaidrotu, kā viedtālruņa lietošana ietekmē tādus parametrus kā enerģijas patēriņš, procesora slodze, grafiskās kartes slodze un atmiņas lietojums. Datu uzkrāšanas testēšanas scenāriji izstrādāti Android viedtālruņu modeļiem, kuri aprīkoti ar mākslīgā intelekta (AI) apakšprocesoriem, piemēram, viedtālruņi ar sekojošām mikroshēmām:

- Kirin 970 (*Kirin 970*, 2017) (Huawei Mate 10, Huawei Mate 10 Pro) vai jaunāki modeļi;
- Exynos 8895 (*Exynos 8895*, 2017) (Samsung Galaxy S8, Samsung Galaxy Note 8) vai jaunāki;
- Snapdragon 845 (*Snapdragon 845*, 2018) (Google Pixel 3) vai jaunāki.

Definēti sekojoši datu uzkrāšanas testēšanas scenāriji:

- dīkstāves režīms – visas lietotnes ir izslēgtas vai ir ieslēgts lidmašīnas režīms;
- pamata darbības režīms – ierīce nav pieslēgta internetam un tiek izmantota tikai pamata darbībām, tai skaitā telefona zvaniem, SMS, kalkulatoram, piezīmju veikšanai;
- interneta pieslēguma režīms – viedtālrunis savienots ar internetu, izmantojot WiFi un/vai mobilo tīklu (4G, LTE), tiek izpildītas dažādas lietotnes, galvenokārt tās, kuras izmanto lietotājs ikdienā: pārlūkprogramma (Google Chrome), sociālie mediji (Facebook, Instagram, Twitter), mobilā banka (Revolut, PayPal, GooglePay), e-pasts (Gmail);
- paplašinātais režīms - iekārta ir savienota ar internetu un tiek izmantotas lietotnes, kuras rada ievērojamu slodzi iekārtai: spēles (populārākās (Google Play, 2021), *Walking Dead* (*Walking Dead*, 2021), FIFA futbols (*FIFA Football*, 2021), *Subway Surfers* (*Subway Surfers*, 2021) u.c.), lietotnes, kas saistītas ar navigāciju (Google Maps) un izklaidi (Youtube, Netflix, Spotify).

Datu uzkrāšanas testēšanas scenārijos tiek uzkrātas sekojošas datu kopas vienības:

- procesora slodze;
- grafiskās kartes slodze;
- operatīvās atmiņas slodze;
- akumulatora uzlādes līmenis;
- procesa piespiedu apturēšanas gadījumi pārslodzes rezultātā;
- renderēto kadru statistika.

Datu uzkrāšanai izmantoti izstrādātāju instrumenti, kuri pieejami Android iekārtās pēc noklusējuma. Kā arī datu uzkrāšanas nolūkam izmantotas speciālas lietotnes, pieejamas Google Play veikalā, piemēram, Activity Monitor: Task Manager (Activity Monitor, 2021), Simple System Monitor (Simple System Monitor, 2021). Datu ieguve projekta ietvaros iekļauj datu kopas noteikšanu un potenciālo datu tipu noteikšanu.

4.3.1. DATU KOPA

Datu kopā iekļauti iepriekš minētie parametri - procesora slodze, grafiskās kartes slodze, operatīvās atmiņas slodze, iekārtas akumulatora uzlādes līmenis, procesa piespiedu apturēšanas gadījumi pārslodzes rezultātā, renderēto kadru statistika. Šajā apakšnodaļā izskaidroti uzskaitīto parametru apraksti.

Procesora slodze

Procesora laiks ir mērvienība, kurā iekārtas procesora bloks (CPU) tiek izmantots lietotnes vai operētājsistēmas instrukciju apstrādei. Lietderīgi mērīt CPU laiku procentos no CPU jaudas, sauktu par CPU slodzi. CPU laikam un CPU slodzei ir divi nozīmīgi pielietojumi - sistēmas vispārējās slodzes kvantitatīva noteikšana un procesora koplietošanas slodžu noteikšana. Ja CPU slodze ir augsta, lietotājs var saskarties ar aizkavi lietotņu izpildē, augsta CPU slodze norāda uz nepietiekamu apstrādes jaudu (White, 2022). Otrs lietojums saistīts ar paralēlu lietotņu izpildes slodžu noteikšanu - kā procesors tiek koplietots starp vairākām vienlaicīgām lietotnēm. Liela CPU slodze vienai lietotnei var norādīt, ka tai ir nepieciešama apstrādes jauda vai arī tā darbojas nekorekti. CPU laiks savukārt ļauj izmērīt vienai lietotnei nepieciešamo apstrādes jaudu (White, 2022). CPU laiku un CPU lietojumu var noteikt vai nu katram pavedienam, katram procesam vai visai sistēmai. Darba ietvaros pielietota procesora procentuālā slodze visai sistēmai.

Grafiskās kartes slodze

Grafikas apstrādes bloks (GPU) ir viedierīces sastāvdaļa, kas paredzēta, lai ātri apstrādātu un mainītu atmiņu un paātrinātu attēlu izveidi izvadīšanai uz displeja. GPU

paralēlā struktūra padara tos efektīvākus par centrālajiem procesoriem (Horowitz, 2021). No lietotāja skatupunkta lietojumprogrammas darbojas ātrāk, ja tās izmanto GPU apstrādes jaudu, lai uzlabotu veiktspēju. GPU sastāv no simtiem mazāku kodolu, šī arhitektūra nodrošina GPU augsto skaitļošanas veiktspēju. Līdzīgi kā CPU lietojums, GPU lietojums tiek mērīts procentos no GPU jaudas.

Operatīvās atmiņas slodze

Operatīvā atmiņa (RAM) ir ātrdarbīga un īslaicīga datu glabāšanas vieta, kuru iekārta izmanto izpildes laikā. RAM ir ātrdarbīga un piemērota informācijas glabāšanai par lietotnēm, kuras noteiktajā brīdī darbojas un to lietotajiem datiem (Villinger, 2019). RAM atmiņas izmantošanas slodzes metriku veido divi mērījumi - kopējā pieejamā atmiņa gigabaitos (GB) un konkrētā laika brīdī izmantotās atmiņas apjoms. Piemēram, 2020. gada mobilajam tālrunim Samsung Galaxy Note 20 Ultra 5G (Gsmarena, 2020) ir pieejama 12 GB RAM atmiņa. Jebkurā brīdī šī iekārta var izmantot jebkuru šīs atmiņas apjomu. Lietošanas gadījuma piemērs - 12 GB pieejamā RAM, lietošanā 8.76 GB, lietojumu var mērīt procentos no RAM ietilpības attiecīgi 73% vai 0,73.

Iekārtas akumulatora uzlādes līmenis

Uzlādes līmenis (SOC) ir akumulatora uzlādes līmenis attiecībā pret tā jaudu, SOC mērvienības ir procentu punkti (0% = tukšs; 100% = pilns). Lietotņu ietekme uz akumulatora darbības laiku ir atkarīga no akumulatora uzlādes līmeņa un ierīces uzlādes stāvokļa (Abdi et al., 2017). Lietotnes ietekme, kamēr iekārta pievienota maiņstrāvas lādētājam, ir niecīga, tādēļ pētījumam pielietotas no maiņstrāvas lādētāja atvienotas iekārtas.

Procesa piespiedu apturēšanas gadījumi pārslodzes rezultātā

Katra Android lietotne darbojas savā procesā, kas turpina izpildi līdz vairs nav nepieciešams un sistēmai jāatbrīvo atmiņa, lai to varētu izmantot citas lietotnes. Būtiska Android iezīme ir tā, ka lietotnes darbības laiku nekontrolē pati lietotne, to nosaka sistēma (Android Developers, 2020). Lietotņu komponentes ietekmē procesu izpildes ilgumu, un, ja šīs komponentes izmantotas neveiksmīgi, sistēma var pārtraukt lietotnes procesu, kamēr tā turpina darboties. Procesu piespiedu pārtraukšanas datu ierakstu biežums parasti ir neliels un tos var sasniegt tikai plaši izmantojot lietotnes.

Renderēto kadru statistika



Lietotāja saskarnes (UI) veiktspēja nodrošina, ka lietotne atbilst tās funkcionālajām prasībām, un mijiedarbība ar lietotni ir vienmērīga, darbojoties ar konsekventiem 60 kadriem sekundē, bez aizkavētiem vai zaudētiem kadriem.

Datu ieguvei uzskaitītajām grupām pētījuma procesā izveidota metodoloģija, lai apkopotu iekārtas veiktspējas datus. Metodoloģija plašāk aprakstīta sekojošajā apakšnodaļā.

4.3.2. DATU IEGUVES METODOLOĢIJA

Pētījuma procesā apskatītas divas datu ieguves metodes - atbalsta lietotnes izstrāde un pieejamo Android profilēšanas rīku izmantošana. Atbalsta lietotnes priekšrocības ir spēja nodrošināt pieeju noteiktu datu uzkrāšanai formātos, kas būtu izmantojami bez iepriekšējas apstrādes, savukārt trūkums ir nepieciešamās paaugstinātās lietotnes privilēģijas un izstrādes izmaksas. Android Studio izstrādātāji piedāvā profilēšanas rīkus, kuri nodrošina gandrīz tūlītējus datus, lai palīdzētu analizēt, kā noteikta lietotne izmanto CPU, GPU, RAM, tīklu un akumulatora resursus. Profilēšanas rīku nepilnība ir tā, ka tie uzrāda datus tikai par vienu lietotni, nevis visu sistēmu. Profilēšanas rīki ir lietderīgi konkrētas lietotnes profilēšanā un optimizēšanā, tomēr tie nesniedz pienācīgu informāciju par visu iekārtu un tās darbības procesiem.

Android viedierīces var profilēt, pielietojot arī sistēmas izsekošanu (tracing). Izsekošanas datņu lielums palielinās salīdzinoši strauji, piemēram, ierakstot viedierīces sistēmas darbības 10 sekundes testa iekārtā ar noklusējuma lietotnēm, nodrošina izsekošanas datni 8,5 MB lielumā, bet izsekošanas datne piecām minūtēm sasniedz 387,8 MB (skat., 4.5.att.). Šīs izsekošanas datnes sniedz datus par CPU lietojumu, CPU frekvenci un uzrāda visus ierīces procesus, savukārt tie nesniedz informāciju par RAM lietojumu un SOC stāvokli.

 trace-exynos9611-QP1A.190711.020-12-40-54.perfetto-trace	8,5 MB
 trace-exynos9611-QP1A.190711.020-12-45-37.perfetto-trace	387,8 MB

4.5.attēls. Android iekārtas izsekošanas datņu apjoms (autora veidots).

Kā alternatīva Android platformā iebūvētajiem monitorēšanas instrumentiem un atbalsta lietotnes izstrādei ir datu ieguves scenārijs, kurā tiek pielietota jau kāda publiski pieejama sistēmas monitorēšanas lietotne ar papildu iespējām saglabāt informāciju kā izvadi turpmākai analīzei. Pieejamas vairākas lietotnes - Cool Tool Pro (*Cool Tool Pro*, 2022), Activity Monitor (*Activity Monitor*, 2021), Simple System Monitor (*Simple System Monitor*, 2021), kuras var veikt sistēmas veiktspējas mērījumus. Gandrīz neviena no

pieejamajām lietotnēm nenodrošina žurnādatņu izveides un eksportēšanas iespējas, izņemot Cool Tool Pro, kura pielietota datu ieguvei.

Datu uzkrāšanai definēti četri testēšanas scenāriji - dīkstāves, pamata darbības, interneta pieslēguma un paplašinātais režīms. Katrs režīms izpildīts trīsdesmit minūtes, katru minūti veicot žurnādatņu ierakstus ar mērījumiem, pielietojot divas viedierīces (Samsung Galaxy A51 Exynos 9611 un Google Pixel 3 Snapdragon 845), iegūtā datu kopa satur 240 ierakstus, to iespējams palielināt veicot žurnādatņu ierakstus katras 10 sekundes.

Definētajā datu ieguves scenārijā ir ievēroti principi - trīsdesmit minūtes veikta lietošanas gadījums, sekojot scenārijam - dīkstāves, pamata darbības, interneta pieslēguma un paplašinātais režīms. Cool Tool Pro uzstādīts testa iekārtās, kas reģistrē CPU, RAM, akumulatora temperatūru, uzlādes stāvokli, kā arī pielietotas monitoringa sistēmas, lai iegūtu CPU un GPU frekvences. Iekārtas testēšanas scenārijos netiek savienotas ar darbstaciju, jo savienojot notiek uzlādes process. Testēšanas scenārijs vienmēr sākts ar pilnībā uzlādētu iekārtas akumulatoru atkārtoti veicot iekārtas akumulatora uzlādi pirms nākamā testēšanas scenārija izpildes. Pēc noteikta testēšanas scenārija iekārtas iestatījumi koriģēti, lai atbilstu nākamajam testēšanas scenārijam un dati no iepriekšējā scenārija apkopoti.

Pētījumā uzmanība pievērsta iekārtas akumulatora lietojumam un, lai iegūtu datus nepieciešams profilēt iekārtas akumulatora lietošanu, ko iespējams veikt izmantojot Android izstrādātāju nodrošinātos rīkus Batterystats un Battery Historian (Android Developers, 2021). Pētījumā iegūtie Batterystats un Battery vēstures dati pievienoti kā vēsturiskās vērtības un iekļauti datu kopā. Akumulatora profilēšanas dati sniedz papildu informāciju, kura nav pieejama lietojot citus rīkus un ļauj analizēt akumulatora enerģijas patēriņu balstoties uz iekārtā notiekošajiem procesiem. Akumulatora lietojumam nav noteiktas mērķa vērtības, ir iespējams analizēt vidējās akumulatora sprieguma vērtības un akumulatora uzlādes līmeni konkrētiem testēšanas scenārijiem. Aptuvenā iekārtas akumulatora enerģijas patēriņa procentuālā vērtība, kura iegūta testēšanas scenārijos pielietojot Battery Historian un Batterystats, iekļauta datu kopā kā prognozējamā mērķa vērtība.

Datu kopa satur sekojošus parametrus (skat VIII pielikumu) - OpenGL Es versiju, Java VM versiju, Android API līmeni, Android versiju, CPU tipu, testēšanas scenāriju, izpildes laiku, vidējo CPU slodzi, izmantoto RAM, procentuālo RAM, akumulatora stāvokli, akumulatora temperatūru, GPU frekvenci, lidmašīnas režīmu, Wi-Fi stāvokli un

pielietoto lietotņu stāvokli. Lai noteiktu pieejamo parametru ietekmi uz iekārtas baterijas akumulatora izlietojumu, veikta datu statistiskā analīze.

4.3.3. STATISTISKĀ ANALĪZE

Statistiskās analīzes mērķis ir noskaidrot, kādi procesi ietekmē viedtelefona akumulatora uzlādes stāvokli. Vienkāršākais veids kā to noskaidrot ir pielietot korelāciju analīzi (Kristapsone, 2019).

Valid cases = 246; cases with missing value(s) = 93.

Variable	N	Mean	S.E. Mean	Std Dev	Variance	Kurtosis	S.E. Kurt	Skewness	S.E. Skew	Range	Minimum	Maximum	Sum
State_of_charge	246	95,33	,24	3,81	14,48	-,36	,31	-,76	,16	14,00	86,00	100,00	23450,00
Battery_temperature	246	28,13	,13	2,09	4,38	-1,15	,31	-,07	,16	7,00	24,00	31,00	6921,00
CPU_frequency	246	1294,50	23,53	369,01	136165,78	-1,27	,31	-,24	,16	1390,00	403,00	1793,00	318448,00
RAM_in_use	246	931,04	6,77	106,16	11269,88	-,04	,31	,13	,16	597,00	659,00	1256,00	229037,00
GPU_load	153	7,46	,98	12,07	145,59	-,63	,39	1,12	,20	32,00	,00	32,00	1141,00

4.6. attēls. Mainīgo aprakstošā statistika (autora veidots).

Pētījumā sagaidīts, ka GPU slodze būs pozitīvi vērsta (skat., 4.6.att.) un akumulatora uzlādes stāvoklis būs negatīvi vērsts (skat., 4.6.att.), jo zināms, ka iekārtu akumulators izlādējas ātrāk, ja iekārta tiek aktīvi izmantota un visvairāk to ietekmē aktīva resursus patērējošu spēļu izpilde, kuras izmanto GPU. GPU slodze un uzlādes stāvokļa novirzes (skat., 4.6.att.) datu kopai nav tuvu nullei, abi mainīgie neatbilst normālsadalījumam. Korelāciju analīzei starp diviem mainīgajiem, kas neatbilst normālsadalījumam, lietderīgāk izmantot Spirmena neparametrisko korelācijas testu (skat., 4.7. att.).

Symmetric measures.

Category	Statistic	Value	Asymp. Std. Error	Approx. T	Approx. Sig.
Ordinal by Ordinal	Spearman Correlation	-,67	,03	-13,97	
Interval by Interval	Pearson's R	-,68	,03	-14,48	
N of Valid Cases		246			

4.7.attēls. Spirmena korelācija starp GPU slodzi un akumulatora uzlādes stāvokli (autora veidots).

Spirmena korelācija uzrāda mērenu negatīvu korelāciju -0,67 starp GPU slodzi un akumulatora uzlādes stāvokli, Pīrsona korelācija ir gandrīz līdzīga vērtībā -0,68 (skat., 4.7. att). Pīrsona pamata korelācijas analīze parāda būtisku negatīvu korelāciju -0,82 starp tiem pašiem mainīgajiem (skat. 4.8. att.).

Correlations		<i>State_of_charge</i>	<i>CPU_frequency</i>	<i>RAM_in_use</i>	<i>RAM_usage</i>	<i>Battery_temperature</i>	<i>GPU_load</i>
<i>State_of_charge</i>	<i>Pearson Correlation</i>	1,00	-,62	,27	,06	-,72	-,82
	<i>Sig. (2-tailed)</i>		,000	,000	,317	,000	,000
	<i>N</i>	246	246	246	246	246	153
<i>CPU_frequency</i>	<i>Pearson Correlation</i>	-,62	1,00	-,06	-,10	,50	,59
	<i>Sig. (2-tailed)</i>	,000		,335	,121	,000	,000
	<i>N</i>	246	246	246	246	246	153
<i>RAM_in_use</i>	<i>Pearson Correlation</i>	,27	-,06	1,00	,81	-,21	,01
	<i>Sig. (2-tailed)</i>	,000	,335		,000	,001	,925
	<i>N</i>	246	246	246	246	246	153
<i>RAM_usage</i>	<i>Pearson Correlation</i>	,06	-,10	,81	1,00	-,05	,22
	<i>Sig. (2-tailed)</i>	,317	,121	,000		,405	,007
	<i>N</i>	246	246	246	246	246	153
<i>Battery_temperature</i>	<i>Pearson Correlation</i>	-,72	,50	-,21	-,05	1,00	,61
	<i>Sig. (2-tailed)</i>	,000	,000	,001	,405		,000
	<i>N</i>	246	246	246	246	246	153
<i>GPU_load</i>	<i>Pearson Correlation</i>	-,82	,59	,01	,22	,61	1,00
	<i>Sig. (2-tailed)</i>	,000	,000	,925	,007	,000	
	<i>N</i>	153	153	153	153	153	153

4.8.attēls. Vienkāršā korelāciju analīze (autora veidots).

Pastāv negatīva korelācija starp akumulatora uzlādes stāvokli, CPU frekvenci, akumulatora temperatūru un GPU slodzi (skat., 4.8. att.). Negatīvā korelācija nozīmē, ka, palielinoties CPU frekvencei, akumulatora temperatūrai vai GPU slodzei, akumulatora uzlādes stāvoklis samazinās. Mašīnmācīšanās algoritma izveides procesā uzskaitītie mainīgie izmantoti kā ievades parametri.

Symmetric measures.					
<i>Category</i>	<i>Statistic</i>	<i>Value</i>	<i>Asymp. Std. Error</i>	<i>Approx. T</i>	<i>Approx. Sig.</i>
Ordinal by Ordinal	Spearman Correlation	-,61	,04	-12,13	
Interval by Interval	Pearson's R	-,62	,03	-12,41	
N of Valid Cases		246			

4.9.attēls. Neparametriskā Spirmena korelācija starp CPU frekvenci un akumulatora uzlādes stāvokli (autora veidots).

Spirmena korelācija parāda vidēji augstu negatīvo korelāciju -0,61 starp CPU frekvenci un uzlādes stāvokli, kā arī līdzīgu vidēji augstu negatīvo Pīrsona korelāciju -0,62 (skat., 4.9. att.), kas nozīmē, ka, palielinoties CPU frekvencei, akumulatora uzlādes stāvoklim ir tendence samazināties.

Symmetric measures.					
<i>Category</i>	<i>Statistic</i>	<i>Value</i>	<i>Asymp. Std. Error</i>	<i>Approx. T</i>	<i>Approx. Sig.</i>
Ordinal by Ordinal	Spearman Correlation	,37	,05	6,17	
Interval by Interval	Pearson's R	,27	,04	4,34	
N of Valid Cases		246			

4.10.attēls. Neparametriskā Spirmena korelācija starp RAM un akumulatora uzlādes stāvokli (autora veidots).

Spirmena korelācija parāda zemu pozitīvu korelāciju 0,37 starp izmantoto RAM un uzlādes stāvokli un zemu pozitīvu Pīrsona korelāciju 0,27 (skat., 4.10. att.). Var secināt, ka RAM izmantošanai ir maza ietekme uz uzlādes stāvokli un RAM būtiski neveicina ierīces akumulatora izlādi.

Symmetric measures.					
Category	Statistic	Value	Asymp. Std. Error	Approx. T	Approx. Sig.
Ordinal by Ordinal	Spearman Correlation	-,71	,04	-15,71	
Interval by Interval	Pearson's R	-,72	,03	-16,31	
N of Valid Cases		246			

4.11.attēls. Neparametriskā Spirmena korelācija starp akumulatora temperatūru un akumulatora uzlādes stāvokli (autora veidots).

Spirmena korelācija parāda vidēji augstu negatīvu korelāciju -0,71 starp akumulatora temperatūru un uzlādes stāvokli, kā arī gandrīz līdzīgu vidēji augstu negatīvu Pīrsona korelāciju -0,72 (skat., 4.11. att.). Var secināt, ka, paaugstinoties akumulatora temperatūrai, uzlādes līmenim ir tendence samazināties.

Akumulatora uzlādes stāvoklis, dalīts testēšanas scenārijos (skat. 4.12. att.), parāda, ka testa scenārijus būtu lietderīgi regulāri mainīt datu ieguves laikā. Uzlādes stāvokļa dalījums atkarībā no lietošanas gadījuma (skat. 4.12. att.) uzrāda, ka akumulatora izlādēšanās nav novērojama, ierīcei atrodoties dīkstāves un pamatdarbības režīmā, uzlādes stāvoklis saglabājas 95% robežās, veicot 122 ierakstus. Interneta savienojuma režīms sāk ietekmēt uzlādes stāvokli un ietekmē akumulatora stāvokļa samazināšanos, sākot no 97% līdz 92% (skat. 4.12. att.). Dati liecina, ka 184 izmērītie ieraksti neietekmē akumulatoru uzlādes stāvokli, tādēļ būtu lietderīgi testēšanas režīmus koriģēt. Paplašinātais režīms ievērojami izlādē ierīces akumulatoru, 62 ieraksti uzrāda iekārtas akumulatora uzlādes stāvokļa izmaiņas no 94% līdz 86%. Dati liecina, ka pastāv korelācija starp testēšanas scenāriju un akumulatora uzlādes stāvokli.

Test case	State of charge										Total					
	86	87	88	89	90	91	92	93	94	95		96	97	98	99	100
Basic operation	,00	,00	,00	,00	,00	,00	,00	,00	,00	6,00	13,00	12,00	23,00	8,00	,00	62,00
	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	9,68%	20,97%	19,35%	37,10%	12,90%	,00%	100,00%
	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	31,58%	68,42%	42,86%	65,71%	28,57%	,00%	25,20%
	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	2,44%	5,28%	4,88%	9,35%	3,25%	,00%	25,20%
Extended use	5,00	6,00	7,00	10,00	10,00	4,00	5,00	7,00	8,00	,00	,00	,00	,00	,00	,00	62,00
	8,06%	9,68%	11,29%	16,13%	16,13%	6,45%	8,06%	11,29%	12,90%	,00%	,00%	,00%	,00%	,00%	,00%	100,00%
	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	62,50%	33,33%	44,44%	,00%	,00%	,00%	,00%	,00%	,00%	25,20%
	2,03%	2,44%	2,85%	4,07%	4,07%	1,63%	2,03%	2,85%	3,25%	,00%	,00%	,00%	,00%	,00%	,00%	25,20%
Idle mode	,00	,00	,00	,00	,00	,00	,00	,00	,00	,00	,00	,00	12,00	20,00	28,00	60,00
	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	20,00%	33,33%	46,67%	100,00%
	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	34,29%	71,43%	100,00%	24,39%
	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	,00%	4,88%	8,13%	11,38%	24,39%
Internet connection	,00	,00	,00	,00	,00	,00	3,00	14,00	10,00	13,00	6,00	16,00	,00	,00	,00	62,00
	,00%	,00%	,00%	,00%	,00%	,00%	4,84%	22,58%	16,13%	20,97%	9,68%	25,81%	,00%	,00%	,00%	100,00%
	,00%	,00%	,00%	,00%	,00%	,00%	37,50%	66,67%	55,56%	68,42%	31,58%	57,14%	,00%	,00%	,00%	25,20%
	,00%	,00%	,00%	,00%	,00%	,00%	1,22%	5,69%	4,07%	5,28%	2,44%	6,50%	,00%	,00%	,00%	25,20%
Total	5,00	6,00	7,00	10,00	10,00	4,00	8,00	21,00	18,00	19,00	19,00	28,00	35,00	28,00	28,00	246,00
	2,03%	2,44%	2,85%	4,07%	4,07%	1,63%	3,25%	8,54%	7,32%	7,72%	7,72%	11,38%	14,23%	11,38%	11,38%	100,00%
	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%
	2,03%	2,44%	2,85%	4,07%	4,07%	1,63%	3,25%	8,54%	7,32%	7,72%	7,72%	11,38%	14,23%	11,38%	11,38%	100,00%

4.12.attēls. Uzlādes stāvokļa ierakstu dalījums atbilstoši testēšanas scenārijam (autora veidots).

Akumulatora temperatūras sadalījums testēšanas scenārijos uzrāda, ka akumulators ir zemākā temperatūrā, kad ierīce ir lidojuma režīmā un pamatdarbības režīmā, dīkstāves režīmā 50% ierakstu ir 25 celsija grādos un akumulatora temperatūra nesasniedz 30 grādus (skat., 4.13.att.). Pamatdarbības režīms nedaudz palielina akumulatora temperatūru, jo 58% ierakstu ir zem 28 grādiem pēc Celsija. Ilgstoša lietošana parāda, ka akumulatora temperatūra 87% reģistrēto gadījumu ir 31 grāds pēc Celsija. Ilgstoša lietošana paaugstina akumulatora temperatūru (skat., 4.13.att.).

Test case	Battery temperature								Total
	24	25	26	27	28	29	30	31	
Basic operation	,00	2,00	7,00	13,00	14,00	18,00	8,00	,00	62,00
	,00%	3,23%	11,29%	20,97%	22,58%	29,03%	12,90%	,00%	100,00%
	,00%	5,71%	33,33%	32,50%	35,00%	50,00%	47,06%	,00%	25,20%
	,00%	,81%	2,85%	5,28%	5,69%	7,32%	3,25%	,00%	25,20%
Extended use	,00	,00	,00	,00	3,00	5,00	,00	54,00	62,00
	,00%	,00%	,00%	,00%	4,84%	8,06%	,00%	87,10%	100,00%
	,00%	,00%	,00%	,00%	7,50%	13,89%	,00%	100,00%	25,20%
	,00%	,00%	,00%	,00%	1,22%	2,03%	,00%	21,95%	25,20%
Idle mode	,00	30,00	4,00	12,00	12,00	2,00	,00	,00	60,00
	,00%	50,00%	6,67%	20,00%	20,00%	3,33%	,00%	,00%	100,00%
	,00%	85,71%	19,05%	30,00%	30,00%	5,56%	,00%	,00%	24,39%
	,00%	12,20%	1,63%	4,88%	4,88%	,81%	,00%	,00%	24,39%
Internet connection	3,00	3,00	10,00	15,00	11,00	11,00	9,00	,00	62,00
	4,84%	4,84%	16,13%	24,19%	17,74%	17,74%	14,52%	,00%	100,00%
	100,00%	8,57%	47,62%	37,50%	27,50%	30,56%	52,94%	,00%	25,20%
	1,22%	1,22%	4,07%	6,10%	4,47%	4,47%	3,66%	,00%	25,20%
Total	3,00	35,00	21,00	40,00	40,00	36,00	17,00	54,00	246,00
	1,22%	14,23%	8,54%	16,26%	16,26%	14,63%	6,91%	21,95%	100,00%
	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%	100,00%
	1,22%	14,23%	8,54%	16,26%	16,26%	14,63%	6,91%	21,95%	100,00%

4.13.attēls. Akumulatora temperatūras ierakstu dalījums atbilstoši testēšanas scenārijam (autora veidots).

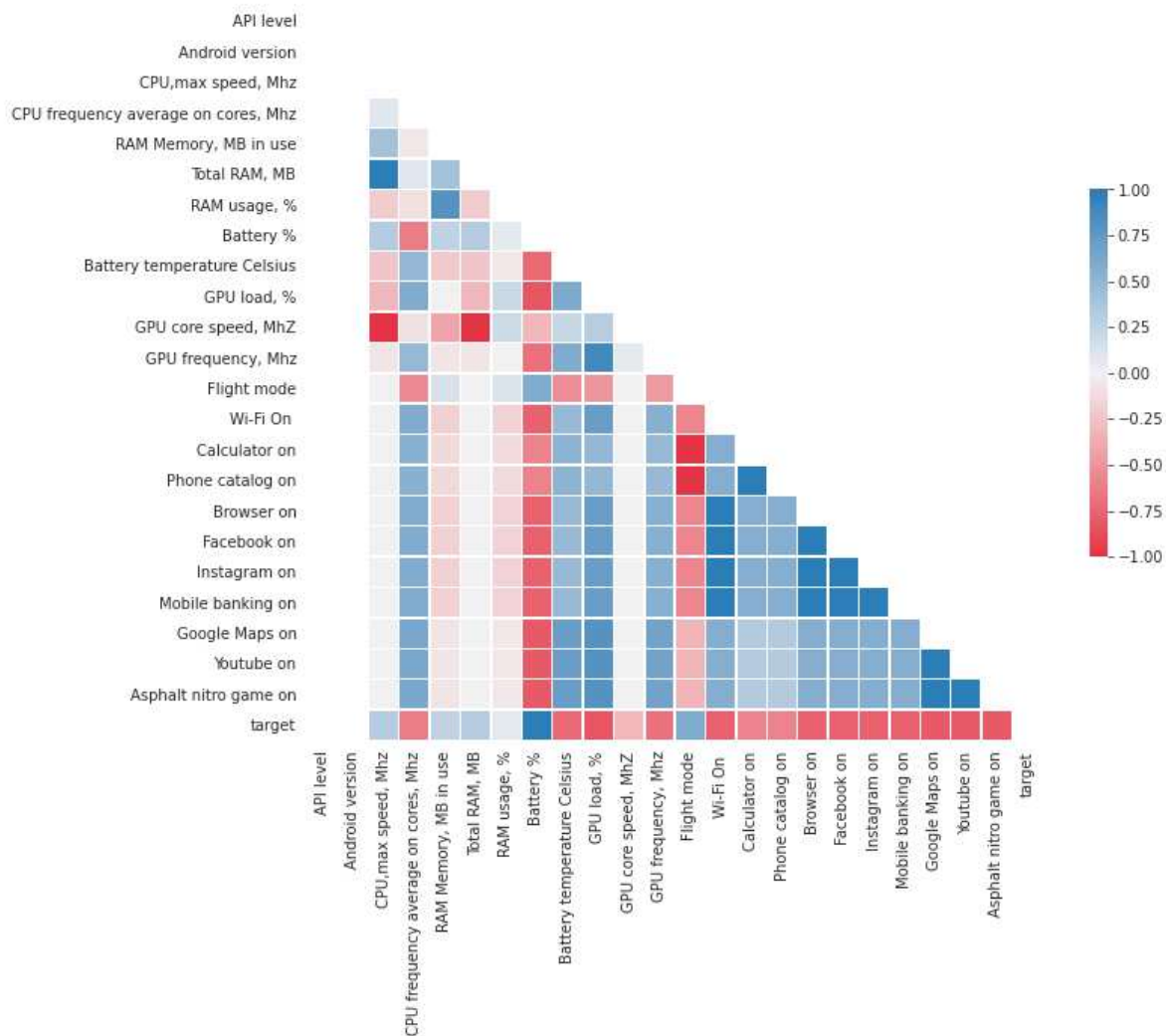
GPU frekvence, dalīta testēšanas scenārijos, uzrāda, ka GPU netiek izmantots, kad ierīce atrodas lidojuma režīmā (100%), un gandrīz netiek izmantota (74,19%), kad tā ir

pamata darbības režīmā (skat. 4.14.att.). Interneta savienojuma testēšanas scenārijā GPU frekvences lietojums palielinās - GPU tiek izmantots 38,79% gadījumu un atrodas dīkstāvē 61,29% gadījumu. Paplašinātais režīms parāda, ka GPU tiek izmantots gandrīz visu laiku - tikai 1,61% ierakstu reģistrēti bez GPU frekvences (skat. 4.14.att.).

Test case	GPU frequency			Total
	Not used	587 MHz	733 MHz	
Basic operation	46,00	8,00	8,00	62,00
	74,19%	12,90%	12,90%	100,00%
	31,72%	15,38%	16,33%	25,20%
	18,70%	3,25%	3,25%	25,20%
Extended use	1,00	31,00	30,00	62,00
	1,61%	50,00%	48,39%	100,00%
	,69%	59,62%	61,22%	25,20%
	,41%	12,60%	12,20%	25,20%
Idle mode	60,00	,00	,00	60,00
	100,00%	,00%	,00%	100,00%
	41,38%	,00%	,00%	24,39%
	24,39%	,00%	,00%	24,39%
Internet connection	38,00	13,00	11,00	62,00
	61,29%	20,97%	17,74%	100,00%
	26,21%	25,00%	22,45%	25,20%
	15,45%	5,28%	4,47%	25,20%
Total	145,00	52,00	49,00	246,00
	58,94%	21,14%	19,92%	100,00%
	100,00%	100,00%	100,00%	100,00%
	58,94%	21,14%	19,92%	100,00%

4.14.attēls. GPU frekvences ierakstu dalījums atbilstoši testēšanas scenārijiem (autora veidots).

Datu kopai ģenerētais korelāciju grafiks ilustrē mašīnmācīšanās algoritmu iespējamās mainīgos (skat. 4.15.att.). Vidēji augsta negatīvā korelācija galvenokārt pastāv ar mainīgajiem — akumulatora temperatūru, GPU frekvenci, CPU frekvenci, kā arī paplašinātā testēšanas scenārija nosacījumiem (ieslēgtās lietotnes un iestatījumi): Wi-Fi, pārlūkprogramma, Facebook, Instagram, Mobilā banka, Google maps, Youtube un spēle Asphalt Nitro (skat. 4.15.att.). Tā kā GPU frekvence un CPU frekvence ir cieši saistīta ar mobilajām lietotnēm, kuras tiek izmantotas interneta pieslēguma testēšanas scenārijā, un paplašinātajā scenārijā, var secināt, ka, GPU un CPU ir mainīgie, kas būtiskāk ietekmē akumulatora uzlādes stāvokli, un citi mainīgie, piemēram, specifiskas lietotnes, nesniegs papildus ieguvumus prognozes rezultātam.



4.15.attēls. Korelāciju grafiks (autora veidots).

Datu kopas identificētie būtiskie mainīgie ir sekojoši: GPU frekvence, CPU frekvence, akumulatora temperatūra, lidojuma režīms, GPU kodola ātrums un izmantotais RAM apjoms. Lai noteiktu piemērotākos mašīnmācīšanās algoritmus pētījumā pielietots scikit-learn (*Scikit Learn*, 2022) ietvars.

GPU frekvence, CPU frekvence, lidojuma režīms un RAM mainīgie no pieejamās datu kopas pētījuma priekšapstrādes procesā pārveidoti algoritma lietošanai. GPU priekšapstrādes rezultātā ir tikai divi stāvokļi — tā vai nu darbojas, vai ir izslēgta, izmainītās vērtības ir divi stāvokļi — 0 un 1 (skat. 4.16.att.). Identiska mainīgā vērtības maiņa pielietota lidojuma režīmam, jo arī tas ir tikai divos stāvokļos - ieslēgts vai izslēgts (skat., IX pielikums).


```
data_test['GPU frequency'].describe()
count      246.000000
mean       270.085366
std        327.638472
min         0.000000
25%        0.000000
50%        0.000000
75%        587.000000
max        733.000000
Name: GPU frequency, dtype: float64
```

4.16.attēls. GPU frekvences mainīgā vērtības (autora veidots).

CPU frekvence un RAM mainīgie priekšapstrādes procesā pārveidoti kvartīļu grupās atbilstoši datu sadalījumam (skat. 4.17.att.). Neizmantojie mainīgie - OpenGL, Java VM, API līmenis, CPU maksimālais ātrums un pārējie, kuri neveidoja būtiskas korelācijas, priekšapstrādes procesā izņemti (skat., IX pielikums).

```
data_test['CPU average'].describe()
count      246.000000
mean       1294.504065
std        369.006474
min         403.000000
25%        989.500000
50%        1271.500000
75%        1678.500000
max        1793.000000
Name: CPU average, dtype: float64
```

4.17.attēls. CPU frekvences mainīgā vērtības (autora veidots).

CPU vērtības dalītas kvartilēs atbilstoši sadalījumam starp pieejamajām vērtībām datu kopā: 1. kvartile 403 - 988, 2. kvartile 989 - 1270, 3. kvartile 1271 - 1677, 4. kvartile 1678 - maksimālā pieejamā vērtība.

Prognozes algoritma mērķa mainīgais ir iekārtas akumulatora enerģijas patēriņa procentuālā vērtība - mainīgais, kura vērtība iegūta pielietojot izstrādātāju rīkus Batterystats un Battery Historian. Pēc datu priekšapstrādes un sagatavošanas tie sadalīti apmācības kopā un testa kopā attiecībā 70/30. Apmācības kopa pielietota, apmācot 70% datu, pēc tam pārbaudot rezultātus ar testa kopas 30%. Algoritma apmācības testēšanai izvēlēti klasifikācijas algoritmi no Scikit learn ietvara - Dummy Classifier, Random Forest Classifier, Linear SVC Classifier, KNeighbors Classifier, Decision Tree Classifier. Pēc

izvēlēto klasifikatoru testēšanas, modeļi ar augstākajiem rādītājiem validēti, izmantojot KFold validāciju (Brownlee, 2020).

4.3.4. ALGORITMU APMĀCĪBAS REZULTĀTI

Dummy Classifier, Random Forest Classifier, Linear SVC Classifier, KNeighbors Classifier, Decision Tree Classifier modeļiem tika pārbaudīti šādi rādītāji - pareizības vērtība (angliski: accuracy score), F1 vērtība (angliski: F1 score), precizitātes vērtība (angliski: precision score), atsaukuma vērtība (angliski: recall score) un klasifikācijas problēmu veiktspējas mērījums (ROC AUC). Lai noteiktu piemērotākos algoritma ievades parametrus izmantota Scikit learn piemērotāko parametru meklēšanas metode GridSearchCV (scikit-learn developers, 2022). Kā bāzes salīdzinājuma vērtība izmantots Dummy klasifikatora modelis, kurš veic prognozi neņemot vērā datu kopas vērtības jeb klasifikators kurš veic vienkārši prognozējamās vērtības minējumu. Šajā daļā tiek uzrādīti tikai bāzes salīdzināmā Dummy klasifikatora rezultāti un augstāko precizitāti uzrādošā Random Forest klasifikatora rezultāti.

Dummy klasifikatora rezultāti

Klasifikatoram piemērotāko parametru atlasei ievadīti sekojoši hiperparametri:

- 'strategy': ('most_frequent', 'stratified', 'prior', 'uniform', 'constant');
- 'random_state': [None, 4, 6, 9];
- 'constant': [0, 1, 2, 3].

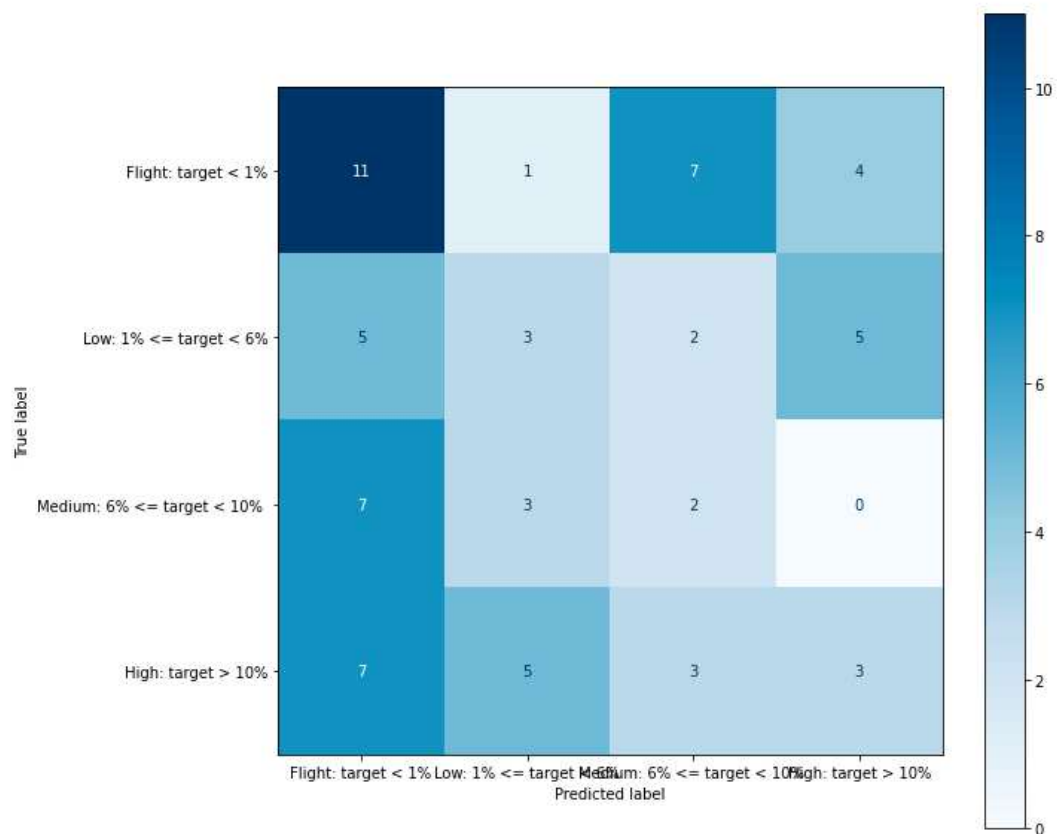
No pārbaudītajiem hiperparametriem Dummy klasifikatora modelim GridSearchCV metode kā piemērotākos uzrāda constant=0, strategy='most_frequent'. Veicot klasifikatora apmācību atgriezti sekojoši rezultāti:

- Precizitāte (accuracy): 0.25;
- Mikro F1 vērtība: 0.25;
- Mikro precizitāte: 0.25;
- Mikro atsaukumi (recall): 0.25;
- Mikro ROC AUC: 0.5;
- Makro F1 vērtība: 0.2398;
- Makro precizitāte: 0.2432;
- Makro atsaukumi (recall): 0.2402;
- Makro ROC AUC: 0.4919;
- Svarota F1 vērtība: 0.2488;
- Svarota precizitāte: 0.2511;

- Svaroti atsaukumi: 0.25;
- Svarots ROC AUC: 0.4874.

	precision	recall	f1-score	support
Flight: target < 1%	0.29	0.22	0.25	23
Low: 1% <= target < 6%	0.20	0.13	0.16	15
Medium: 6% <= target < 10%	0.20	0.25	0.22	12
High: target > 10%	0.31	0.44	0.36	18
accuracy			0.26	68
macro avg	0.25	0.26	0.25	68
weighted avg	0.26	0.26	0.26	68

4.18.attēls. Dummy klasifikatora rezultātu atskaite (autora veidots).



4.19.attēls. Dummy klasifikatora rezultātu matrica (autora veidots).

Dummy klasifikatora precizitāte ir 0,26 (skat.. 4.18.att.), 11 no lidojuma testēšanas akumulatora patēriņa prognozēm bija patiesi pozitīvas ar 26 patiesi negatīvām, 19 viltus pozitīvām un 12 viltus negatīvām prognozēm. Interneta pieslēguma testēšanas scenārija

prognozes uzrādīja 2 patiesus pozitīvus, 44 viltus negatīvus, 12 viltus pozitīvus un 10 viltus negatīvus (skat., 4.19.att.). Viszemākie rādītāji konstatēti pamata darbības testēšanas scenārijā, kuri uzrādīja precizitāti 0,20, atsaukumu 0,13 un F1 0,16 (skat., 4.18.att.), savukārt kopējā klasifikatora precizitāte ir 0,25, var secināt, ka Dummy klasifikators korekti spēj prognozēt katru ceturto rādījumu.

Random Forest klasifikatora rezultāti

Klasifikatoram piemērotāko parametru atlasei izvirzīti sekojoši hiperparametri:

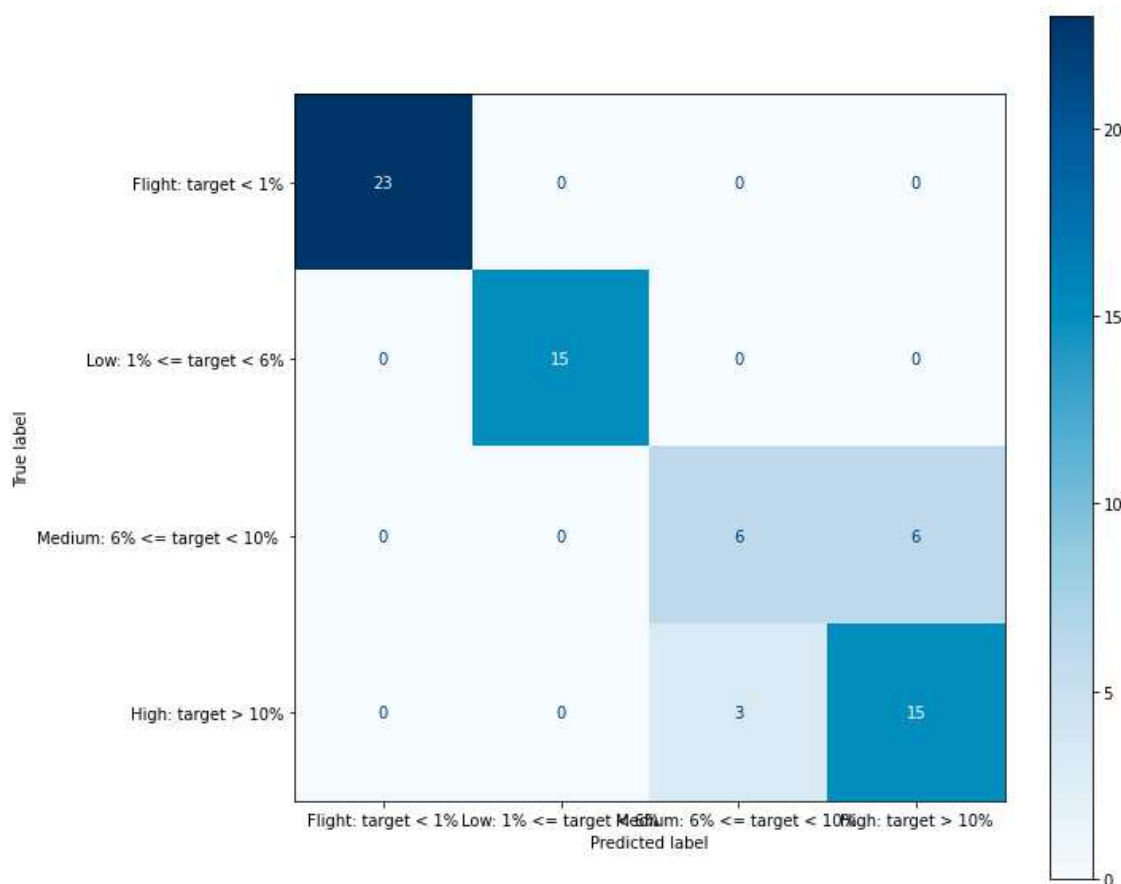
- 'n_estimators': [4, 6, 9, 12];
- 'max_features': ['log2', 'sqrt', 'auto'];
- 'criterion': ['entropy', 'gini'];
- 'max_depth': [2, 3, 5, 10, 15];
- 'min_samples_split': [2, 3, 5];
- 'min_samples_leaf': [1, 5, 8].

No pārbaudītajiem hiperparametriem Random Forest klasifikatora modelim GridSearchCV metode kā piemērotākos uzrāda max_depth=10, max_features='sqrt', min_samples_split=3, n_estimators=9. Veicot klasifikatora apmācību atgriezti sekojoši rezultāti:

- Precizitāte (accuracy): 0.8676;
- Mikro F1 vērtība: 0.8676;
- Mikro precizitāte: 0.8676;
- Mikro atsaukumi (recall): 0.8676;
- Mikro ROC AUC: 0.9313;
- Makro F1 vērtība: 0.8351;
- Makro precizitāte: 0.8452;
- Makro atsaukumi (recall): 0.8333;
- Makro ROC AUC: 0.9220;
- Svarota F1 vērtība: 0.8632;
- Svarota precizitāte: 0.8655;
- Svaroti atsaukumi: 0.8676;
- Svarots ROC AUC: 0.9321.

	precision	recall	f1-score	support
Flight: target < 1%	1.00	1.00	1.00	23
Low: 1% <= target < 6%	1.00	1.00	1.00	15
Medium: 6% <= target < 10%	0.67	0.50	0.57	12
High: target > 10%	0.71	0.83	0.77	18
accuracy			0.87	68
macro avg	0.85	0.83	0.84	68
weighted avg	0.87	0.87	0.86	68

4.20.attēls. Random Forest klasifikatora rezultātu atskaite (autora veidots).



4.21.attēls. Random Forest klasifikatora rezultātu matrica (autora veidots).

Random Forest klasifikatora precizitāte ir 0,87 (skat. 4.20.att), visas 23 lidojuma testēšanas scenārija akumulatora patēriņa prognozes ir patiesi pozitīvas ar 45 patiesi negatīvām, bez viltus pozitīvām un bez viltus negatīvām (skat., 4.21.att.). Interneta pieslēguma testēšanas scenārija prognozes uzrādīja sešus patiesi pozitīvus, 53 viltus negatīvus, trīs viltus pozitīvus un sešus viltus negatīvus (skat., 4.21.att), šī testēšanas scenārija rezultāti uzrādīja precizitāti 0,67, atsaukumu 0,5 un F1 rādītāju 0,57 (skat.,

4.20.att.). Interneta pieslēguma režīma akumulatora patēriņa prognozēm ir zemāka precizitāte nekā pārējiem testēšanas scenārijiem gadījumiem ar precizitāti 67% (skat., 4.20.att.). Kopējā Random Forest klasifikatora precizitāte ar pielietoto datu kopu ir 0,87, kas ir 87% (skat. 4.20.att.), lai arī šāds algoritma rezultāts uzrāda augstu precizitāti, tā iemesls var būt datu kopas relatīvi nelielais apjoms un datu pārmērīga priekšapstrāde, kas izvirza prasību lielākas datu kopas izveidei. Pētījuma sekojošās aktivitātes iekļāva prasību pievienot algoritmam papildu prognozi, kas varētu prognozēt, vai kāda noteikta lietotnes izmantošana ietekmē gala lietotāja pieredzi veidā, kura būtu lietotājam redzama.

4.3.5. PAPILDU PROGNOZES PIEVIENOŠANA

Pētījuma otrajā posmā kā mērķis noteikta papildu prognozes pievienošanu, lai noteiktu, vai konkrēta lietotne ietekmē gala lietotājam renderētos un redzamos kadrus. Akumulatora enerģijas patēriņa līmeņa tēma apskatīta Kang et al. pētījumā (Kang et al., 2011), kurā tiek aprakstīts, kā mobilo ierīču lietošanas paradumi izpaužas kā akumulatora enerģijas patēriņš. Pētnieki galvenokārt koncentrējas akumulatora enerģijas patēriņa prognozēšanai un optimizācijai, savukārt neaplūko citus iekārtas parametrus. Citā pētījumā (Zhang et al., 2013) tiek pārbaudīta to mobilo lietotņu efektivitāte, kas izmanto daudzkodolu procesorus. Daudzkodolu procesori ir nozīmīgi lietotāja saskarnes efektivitātes novērtējuma kontekstā, jo papildus funkcionalitātei lietotnei jānodrošina vienmērīga mijiedarbība ar lietotāju, kas nozīmē, ka tai jādarbojas ar ātrumu vismaz 60 kadri sekundē (FPS), tātad noteikts kadrs lietotājam ir jāparāda aptuveni ik pēc 16 ms. Šis nosacījums aprakstīts vairākās publikācijās (Gómez et al., 2016; Kaasila et al., 2012), kurās pētnieki norāda, ka tādi faktori kā saskarnes palēnināšanās, kadru zaudēšana, lietotnes izpildes aizkaves vai kļūdu rašanās ietekmē gala lietotāja iespaidu par lietotni. Līdzīgi pētījumi, kuros aprakstīti procesi iekārtas akumulatora izmantošanas prognozēšanai veikti arī iepriekš, piemēram, Mansouri et al (Mansouri et al., 2017) un Jia L. (L. Jia, 2017). Mansouri et al. apskata dronus, kuru lidojuma laika pagarināšana saistīta ar laika prognozēšanu, kad akumulators vairs nespēs nodrošināt lidojumu. Autora pētījumā papildus prognozes mērķis ir noteikt, vai lietotne var palēnināties tik ļoti, lai to pamanītu lietotājs, t.i., vai nākamais kadrs tiks parādīts retāk nekā ik pēc 16 ms.

Lietotāja saskarnes veiktspējas testi ļauj pārliecināties, ka lietotne ne tikai atbilst tās funkcionālajām prasībām, bet arī to, ka lietotāja mijiedarbība ar lietotni ir vienmērīga, darbojoties ar konsekventiem 60 kadriem sekundē, bez zaudētiem vai aizkavētiem kadriem. Dumpsys (Developers Android, 2020) ir Android platformā iekļauta

utilītprogramma, kas darbojas iekārtā un atgriež informāciju par sistēmas pakalpojumu statusu. Komandas `gfxinfo` nodošana `dumpsys` nodrošina izvadi ar veikspējas informāciju par iekārtas animācijas kadriem (Developers Android, 2020). UI veikspējas testēšanas procesu var pielietot, izmantojot komandrindas rīkus no darbstacijas ar pievienotu Android ierīci ar priekšnosacījumu, ka ierīcē ir iespējoti izstrādātāja iestatījumi. `Dumpsys` izvada informāciju ar kadru statistiku, kura tiek izdrukāta apkopotā formātā (skat., 4.22.att.). Šie UI veikspējas dati parāda lietotnes kadru renderēšanas veikspēju, kā arī tās stabilitāti vairākos kados.

```
Stats since: 752958278148ns
Total frames rendered: 82189
Janky frames: 35335 (42.99%)
90th percentile: 34ms
95th percentile: 42ms
99th percentile: 69ms
Number Missed Vsync: 4706
Number High input latency: 142
Number Slow UI thread: 17270
Number Slow bitmap uploads: 1542
Number Slow draw: 23342
```

4.22.attēls. `Dumpsys` UI atgrieztie dati (autora veidots).

Pētījumā uzmanība pievērsta tiem kadriem, kuri pārsniedz renderēšanas laiku, īpaši nokavēti renderētajiem kadriem (angliski: `janky frames`), kuri radušies noteiktajā testēšanas scenārijā (skat., 4.22.att.). Šo zaudēto kadru summa iegūta atkārtojot testēšanas scenārijus un izpildes laikā katrā intervālā eksportējot `dumpsys` utilītprogrammas atgrieztos datus `csv` formātā. Datu kopai papildus pievienoti katra intervāla kopējais, nokavētais un nerenderētais kadru skaits. Kopējais kadru skaits ir noteiktajā laika intervālā lietotājam pavisam renderēto kadru skaits. Nokavētais kadru skaits ir to kadru skaits, kuri renderēti pārsniedzot 16ms intervālu, savukārt nerenderētais kadru skaits ir tie kadri, kuri aizņēmuši pārāk daudz laika un lietotājam nav parādīti (skat., 4.23.att.) (*Slow Rendering*, 2021).

Total frames	Janky frames	Missed frames
8750	545	338
8748	545	338
8752	547	340
8764	554	347
8750	546	338
8749	545	338
8736	538	331
8751	548	341
8785	553	340
8751	548	340
8748	544	337
8748	544	337
8749	546	339
8750	546	339
8748	545	338
8748	544	337
8866	629	369
8758	550	343
8750	547	338
8768	555	348
8796	551	341

4.23.attēls. Apkopotās renderēto kadru kopas paraugs (autora veidots).

Pētījuma otrajā posmā testēšanas scenāriji atkārtoti, tos papildinot ar lietotāju saskarnes datiem katrā intervālā (skat., 4.23.att.). Korelāciju analīze uzrāda augstu pozitīvu korelāciju starp iekārtas enerģijas patēriņu un kopējo renderēto kadru skaitu 0,63 un nokavēto kadru skaitu 0,7 (skat., 4.24.att.). Tas nozīmē, ja iekārta patērē vairāk akumulatora enerģijas, tiek renderēts arī lielāks kadru skaits. Otrā pētījuma posma algoritma uzdevums ir prognozēt, vai kāds izmantošanas scenārijs palēninās kadru renderēšanu tādā apmērā, ka vismaz viena kadra pārzīmēšana nākamās minūtes laikā aizņems vairāk nekā 16 ms un būs redzama lietotājam.

		Device_power_estimate_percentage	Total_frames	Janky_frames	Missed_frames
Device_power_estimate_percentage	Pearson Correlation	1,00	,63	,70	,37
	Sig. (2-tailed)		,000	,000	,000
	N	246	246	246	246
Total_frames	Pearson Correlation	,63	1,00	,86	,78
	Sig. (2-tailed)	,000		,000	,000
	N	246	246	246	246
Janky_frames	Pearson Correlation	,70	,86	1,00	,62
	Sig. (2-tailed)	,000	,000		,000
	N	246	246	246	246
Missed_frames	Pearson Correlation	,37	,78	,62	1,00
	Sig. (2-tailed)	,000	,000	,000	
	N	246	246	246	246

4.24.attēls. Korelāciju analīze kadru statistikai un akumulatora patēriņam (autora veidots).

Uzkrātie dati liecina, ka kadri tiek kavēti arī tad, ja iekārtai ieslēgts lidojuma režīms, no datiem ir zināms, ka, ja algoritms paredzētu nokavētos kadrus, tad tas nebūs pārlicinošs rādītājs tam, vai lietotājs redz nekorektu vizualizāciju. Nokavēti kadri viennozīmīgi nenozīmē, ka lietotājam būs redzama nekorekta vizualizācija, tie atspoguļo problemātisko kadru skaitu. Testēšanas scenāriji uzrādīja, ka visos gadījumos iekārtas spēj renderēt atbilstošo kadru skaitu, tomēr paplašinātajā scenārijā bija lielāks nokavēto kadru skaits. Pētījumā ieviests nosacījums, ja kopējais nokavēto kadru procentuālais daudzums ir 15% vai lielāks no visiem, tad iekārtas kadru renderēšanu var uzskatīt par palēninātu un lietotāji to varēs novērot. Nokavēto kadru procentuālā vērtība priekšapstrādes procesā pārveidota par klasifikācijas uzdevumu, lai to savienotu ar pētījuma pirmajā posmā izveidoto algoritmu.

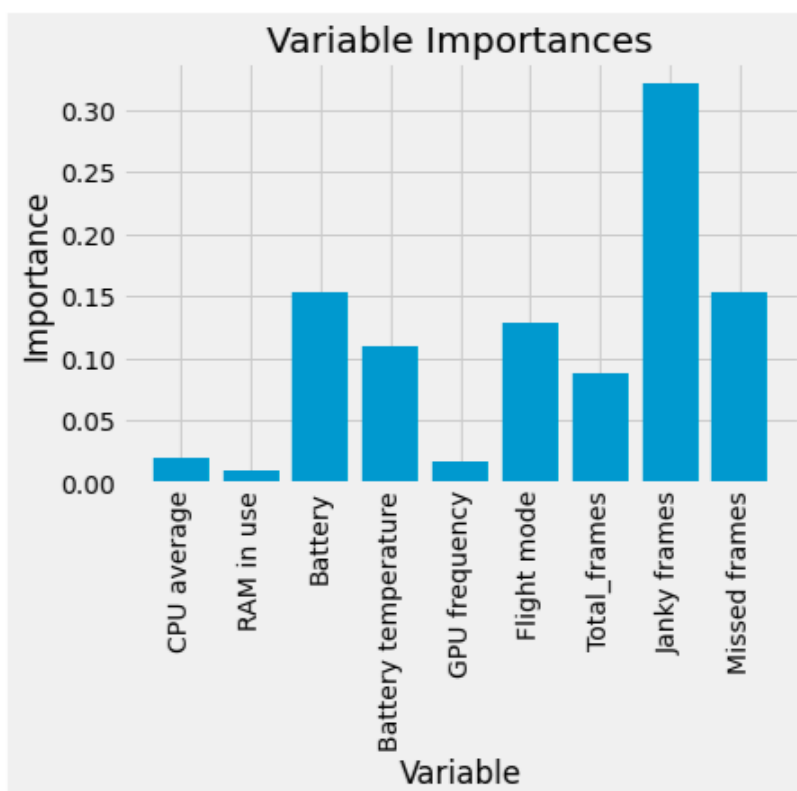
Pētījuma pirmajā posmā Random Forest klasifikators uzrādīja labākos rādītājus, otrajā posmā šim modelim pievienoti iekārtas saskarnes dati. Korelāciju analīze uzrādīja (skat., 4.24.att.) augstu pozitīvu korelāciju starp akumulatora enerģijas patēriņu un renderēto kadru skaitu, tādēļ tika sagaidīts, ka pievienotie parametri ietekmēs modeļa prognozes precizitāti. Pielāgojot Random Forest klasifikatoru un tam pievienojot otru prognozējamo vērtību tika izmantota MultiOutputClassifier (*Multiclass and Multioutput Algorithms*, 2021) no scikit learn ietvara, saglabājot klasifikatora hiperparametrus. Tāpat kā pētījuma pirmajā posmā modelim tika pārbaudīti rādītāji - pareizības vērtība (accuracy score), F1 vērtība (F1 score), precizitātes vērtība (precision score), atsaukuma vērtība (recall score) un klasifikācijas problēmu veiktspējas mērījums (ROC AUC). Klasifikators uzrādīja sekojošus rezultātus:

- Precizitāte (accuracy) akumulatora patēriņš: 0.9797, kavētie kadri: 0.9932
- Micro ROC AUC akumulatora patēriņš: 0.9865, kavētie kadri: 0.9964
- Mikro F1 vērtība: akumulatora patēriņš: 0.9797
- Mikro F1 vērtība: kavētie kadri: 0.9932
- Mikro precizitāte akumulatora patēriņš 0.9797
- Mikro precizitāte kavētie kadri: 0.9932
- Mikro atsaukumi (recall) akumulatora patēriņš: 0.9797
- Mikro atsaukumi (recall) kavētie kadri: 0.9932
- Mikro ROC AUC akumulatora patēriņš: 0.9864
- Mikro ROC AUC kavētie kadri: 0.9964

Klasifikators atgriezta prognozētās akumulatora patēriņa mērķa vērtības ar lielāku precizitāti, kā pirmajā posmā (precizitāte 0,98 jeb 98%). Papildus pievienoto kadru

statistikas parametri ietekmēja prognozes, jo tām bija pozitīva korelācija ar enerģijas patēriņu. Sākotnējā precizitāte nokavētajiem kadriem bija 0,99 vai 99% precizitāte, kas liecina par pārāk mazu datu kopu un/vai datu pārmērīgu priekšapstrādi.

Lai uzlabotu algoritma precizitāti un no datu kopas izņemtu mainīgos, kuri nerada ieguvumus vērtību prognozēšanā, tika noteikta mainīgo ietekme (skat., 4.25.att.). Kavētajiem, kopējiem un nerenderētajiem kadriem pastāv pozitīva augsta korelācija ar iekārtas enerģijas patēriņu, rezultātā tie uzlaboja modeļa precizitāti (skat., 4.25.att.).



4.25.attēls. Datu kopas mainīgo ietekme (autora veidots).

Kā redzams 4.25. attēlā, nozīmīgākie mainīgie ir akumulatora uzlādes līmenis, kavēto kadru skaits, kopējo kadru skaits, nerenderēto kadru skaits un iekārtas lidojuma režīms. Pētījuma pirmajā posmā no datu kopas tika izņemti tie mainīgie, kuri neietekmēja modeļa precizitāti. Pētījuma izstrādes procesā izpratnes palielināšanai ģenerēts vizualizēts algoritma lēmumu koks, attēlojot vienu no Random Forest klasifikatora kokiem, kuru kopējais skaits, pielietojot GridSearchCV noteikts deviņi (skat., X pielikums).

Šāda prognozes vērtība nozīmē ka modelis visticamāk ir pārmērīgi pielāgots konkrētajai datu kopai, nevis to, ka tas uzrāda augstu precizitāti. Modeļa apmācībai ir nepieciešama lielāka datu kopa, iekļaujot datus no vairāk iekārtām un palielinot kopas

izmēru. Pētījuma kopējie rezultāti nav viennozīmīgi interpretējami to esošajā stāvoklī bez lielākas datu kopas un atkārtotas modeļa apmācības. Pēc modeļa apmācības ar lielāku datu kopu un rezultātu verifikācijas un validācijas to var pielietot, lai prognozētu vai lietotne, kura izmanto mobilās iekārtas resursus ietekmēs mobilo iekārtu tādā apjomā, lai to novērotu lietotājs.

Tabulā (skat., 4.4.tabula) sniegts kvantitatīvo mērījumu apkopojums UIS-KDF ietvara aprobācijai viena algoritma apmācības procesā. Pielietotas ietvara kvantitatīvai novērtēšanai izvirzītās pazīmes, kuras apkopotas 3.2.tabulā, pielāgojot noteiktajam aprobācijas scenārijam, izslēdzot no novērtēšanas slāņus, kuri aprobācijas scenārijā netiek pielietoti. Šie rādītāji sniedz atbalstu uzņēmuma lēmumu pieņēmējiem un ieinteresētajām pusēm novērtēt zināšanu izguves ietvara aprobāciju viena algoritma apmācības procesam un pieņemt lēmumus par tā pielietošanu un pilnveidošanu.

4.4. tabula. UIS-KDF viena modeļa apmācības izvērtējuma apkopojums (autora apkopota, Monetization Solutions Inc. dati)

Slānis	Metrika	Mērījums
Tehnoloģiju slānis	Darbspējas laiks	99.98% gada griezumā.
	Datu apstrādes ātrums	273ms - 970ms atkarībā no klienta fiziskās atrašanās vietas.
Publisko lietotņu slānis	Lietotāju pieņemšanas līmenis	Neitrāls. Lietotāji apmācīto algoritmu nepielieto. Tiek pielietots uzņēmuma izstrādātā risinājuma ietekmes novērtēšanai.
	Sistēmas atbildes ātrums	273ms - 970ms atkarībā no klienta fiziskās atrašanās vietas.
Mašīnmācīšanās slānis	Modeļu precizitāte	87% pirmajai prognozei, 99% otrajai prognozei. Liecina par pārāk mazu datu kopu un/vai datu pārmērīgu priekšapstrādi. Apmācību nepieciešams atkārtot ar lielāku datu kopu.
	Apmācības laiks	1 diena katrai apmācības iterācijai

Slānis	Metrika	Mērījums
	Prognozes latentums	400ms apmācītajam algoritmam uzstādītam un lietotam kā klāstera konteinerim
Ietekmējošo kritēriju slānis	Peļņa no kapitāla ieguldījuma	0%, apmācītais algoritms, netiek pielietots uzņēmuma pamatdarbības nodrošināšanai
	Uzņēmējdarbības ietekmes vērtējums	Pozitīvs. Sniedz iespēju izvērtēt uzņēmuma izstrādātā risinājuma ietekmi uz iekārtas resursu patēriņu

4.4. NODAĻAS KOPSAVILKUMS UN SECINĀJUMI

Izstrādātā UIS-KDF ietvara aprobācija jaunuzņēmuma darbībā uzrādīta darba ceturtajā daļā. 4.5.tabulā uzrādīts kā UIS-KDF ietvara raksturiezīmes aprobētas tehnoloģiju ietvaros.

4.5. tabula. UIS-KDF ietvara aprobācija tehnoloģiju ietvaros (autora veidota)

Pazīme	Tehnoloģiju ietvars
Modulārs	Pielietots dalījums piecos slāņos - publisko lietotņu, pārvaldes lietotņu, MM, platformas tehnoloģiju un ietekmējošo kritēriju. Izceļams pielietojums mobilo lietotņu nozarē, kur pielietojot UIS-KDF ietvaru izstrādāts tehnoloģiju ietvars, kurš tiek pielietots jaunuzņēmuma darbības nodrošināšanai.
Izstrādātāji abstrahēti no datu apstrādes platformu sarežģītības	Datu uzglabāšanas un datu apstrādes platformas atdalītas no lietojumprogrammu izstrādes. Šāds izstrādātāju abstrahēšanas veids kā lietderīgs izstrādes ātruma palielināšanas elements pielietots ietvara izstrādē mobilo lietotņu nozarē un mobilo iekārtu bateriju prognozes algoritmu izstrādē.
Dalīta infrastruktūra	Horizontāli un vertikāli mērogojama. Aprobācijas veidi uzrādīja tehnoloģiju

Pazīme	Tehnoloģiju ietvars
	ietvara infrastruktūras elastību regulāri mainīgu prasību nodrošināšanai jaunuzņēmuma darbības atbalstam.
Datu priekšapstrādes modulis	Iekļauts platformas tehnoloģiju slānī. Datu priekšapstrāde mobilo lietotņu un bateriju slodzes testu aprobācijas scenārijiem sagādāja izaicinājumus datu apkopošanā no dažādiem ieguves avotiem.
Rezultātu apstrādes un izplatīšanas modulis	Nodrošināts pielietojot lietojumprogrammu saskarni un trešo pušu savietojamas lietotnes. Rezultātu apstrādes un izplatīšanas moduļu prasības nepieciešams definēt atbilstoši lietošanas gadījumam un klientu prasībām.
Savienojami programmēšanas mezgli	Izmantotas REST un GraphQL lietojumprogrammu saskarnes.
Augstas veiktspējas mērogojama infrastruktūra	Realizētais scenārijs - automātiski mērogojamas infrastruktūras pielietošana izmantojot Google Cloud Platform un Amazon AWS.
Zināšanu izguves atbalsts	Zināšanu izguve izstrādāta divos aprobācijas projektos - pielietoti TensorFlow un Scikit learn izstrādes ietvari.
Lielo datu platformu atbalsts	Iekļauts platformas tehnoloģiju slānī. Lielo datu platformu atbalsts nodrošināts pielietojot Google Cloud un Amazon AWS.
Tiešsaistes analītikas modulis	Iekļauts mobilo lietotņu nozares aprobācijas piemērā, nodrošina platformas tehnoloģiju slānis, aprobācijās tiek aktīvi pielietots biznesa procesos.
Mērķauditorija	UIS-KDF ietvara aprobācija tehnoloģiju ietvaros pielietota jaunuzņēmuma infrastruktūras un biznesa mērķu nodrošināšanai.

Zināšanu izguvi aktīvajā mācību ekosistēmā var risināt, vācot un analizējot datus no konkrētās mācību vides. Apkopotos lietotāju reakcijas datus var izmantot, lai labāk izprastu auditoriju, gūtu ieskatu un veicinātu aktīvu atgriezenisko saiti ar studentiem.

UIS-KDF ietvara aprobācija aktīvās mācīšanās atbalstam izstrādāta teorētiskā kontekstā, ieviešot autora docētajosursos kursos projektu bāzētu aktīvās mācīšanās metodi.

Lai palielinātu spēlētāja vērtību un samazinātu spēlētāja ieguves izmaksas, zināšanu izguve, uzkrājot un analizējot datus no spēlēm tādā veidā, kas atbilstu VDAR, jāīsteno tā, lai tā negatīvi neietekmētu spēlētājus. Pētījuma procesā izstrādātais UIS-KDF ietvars aprobēts izmantošanai mobilo lietotņu nozarē, izveidojot infrastruktūru un mobilās pakotnes, kā arī pārbaudot ietvara darbību uzstādot lietotnes publiski pieejamās mobilajās lietotnēs.

UIS-KDF aprobācija jaunuzņēmuma darbības nodrošināšanai vienkāršoja uzņēmuma infrastruktūras izstrādes procesu, modulārais dalījums slāņos spēj nodrošināt globālu mērogojamību uzņēmumam paplašinot un diversificējot darbību. UIS-KDF pielietošana sniedza iespēju veikli sagatavot uzņēmumu SOC-2 sertifikācijas procesam un iekšējo procesu sagatavošanā VDAR atbilstībai. UIS-KDF modularitāte nodrošina mazāko tiesību principu (angliski: principle of least privilege), kā arī izstrādātājiem sniedz iespējas lietot tiem zināmās tehnoloģijas abstrahējoties no kopējās infrastruktūras sarežģītības (Monetization Solutions Inc, 2022).

Darba ceturtajā daļā:

- izstrādātas UIS-KDF ietvara aprobācijas un pielietojuma paraugi;
- ir aprakstīts UIS-KDF aprobācijas pielietojums aktīvās mācīšanās atbalstam darba autora pasniegtajos lekcijuursos;
- ir apkopota un aprakstīta UIS-KDF ietvara aprobācija jaunuzņēmuma Monetization Solutions Inc. darbības nodrošināšanai, uzrādot UIS-KDF ietvara aprobācijas izvērtējumu;
- ir detalizēti aprakstīti UIS-KDF ietvara mašīnmācīšanās slāņa viena algoritma apmācības process, iekļaujot detalizētu procesa izklāstu;
- ir detalizēti uzrādīti kā UIS-KDF ietvars aprobācijas procesā tiek pielietots izstrādājot tehnoloģiju ietvaru zināšanu izguves sistēmai.

GALVENIE REZULTĀTI UN SECINĀJUMI

Promocijas darba mērķis - izstrādāt konceptuālu zināšanu izguves ietvaru no viedo sistēmu lietotāju reakcijas datiem (UIS-KDF), lai, pielietojot šo ietvaru un tā izmantošanas metodoloģiju, sniegtu atbalstu elastīgu un mērogojamu intelektuālu sistēmu ar zināšanu izgūšanas spēju izstrādei, ir sasniegts.

Izpildītie darba mērķa sasniegšanai izvirzītie uzdevumi:

- ir veikta pētniecības uzskatu par zināšanām, viedajām sistēmām un lietotāju reakcijas datiem apkopošana un analīze. Rezultātu apkopojums uzrādīts darba teorētiskajā daļā;
- identificēti un analizēti pieejamie zināšanu izguves ietvari un to tehnoloģiju aprobācija;
- identificēti būtiskie zināšanu izguves ietvaru arhitektūras elementi un ietekmējošie kritēriji. Identificētie elementi un komponentes iekļautas UIS-KDF ietvarā;
- izstrādāts konceptuāls zināšanu izguves ietvars - UIS-KDF. UIS-KDF ietvars pielietots jaunuzņēmuma tehnoloģisko procesu izveidei;
- izstrādāta UIS-KDF ietvara novērtēšanas pieeja un apkopota novērtēšana aprobācijām;
- izstrādāti UIS-KDF ietvara aprobācijas un pielietojuma paraugi. Divi no izstrādātajiem paraugiem izmantoti jaunuzņēmuma darbības nodrošināšanai.

Darba teorētiskā vērtība - izveidots un aprobēts konceptuāls zināšanu izguves ietvars no viedo sistēmu lietotāju reakcijas datiem (UIS-KDF), iekļaujot tehnoloģiskos aspektus.

Praktiskais jaunievedums - UIS-KDF ietvars izstrādāts, lai atvieglotu zināšanu izguves sistēmas izstrādes agru posmu un risina sarežģījumus, kas saistīti ar arhitektūras plānošanu un infrastruktūras izveidi. UIS-KDF jaunievedums ir tā pielāgošana dažādiem organizatoriskiem kontekstiem. Ietvara vispārējā arhitektūra to ļauj pielāgot, pamatojoties atbilstoši noteiktās nozares prasībām un darbības mērķiem. Elastība ir nozīmīga, jo ļauj organizācijām izveidot zināšanu izguves sistēmu tā, lai tā atbilstu organizācijas specifiskajiem procesiem un stratēģiskajiem mērķiem. Izstrādāti UIS-KDF ietvara aprobācijas piemēri un UIS-KDF ietvars aprobēts jaunuzņēmuma darbības nodrošināšanai.

Zinātniskais jaunievedums - izpētīti un salīdzināti zināšanu izguves ietvari, identificēti to trūkumi un risinājumi iekļauti UIS-KDF ietvara izveidē. Izveidota saistošo pētījumu atlases pieeja un meklēšanas stratēģija, kura apkopo detalizētu informāciju par zināšanu izguvi, identificējot nozīmīgākos pētniekus, sfēras un nozīmīgākās publikācijas. Izstrādāts UIS-KDF zināšanu izguves ietvars no viedo sistēmu lietotāju reakcijas datiem, kas balstīts uz promocijas darbā apskatīto zināšanu izguves procesu. Izstrādāta adaptējama UIS-KDF ietvara arhitektūra, kas balstās uz mērogojamu mākoņskaitļošanas tehnoloģiju izmantošanu. Izstrādāta UIS-KDF pielietošanas metodoloģija un pielietošanas ierobežojumi. Izstrādāta UIS-KDF ietvara pielietojuma kvantitatīvo rādītāju novērtējuma pieeja.

Pētījumā izvirzītās tēzes tikušas apstiprinātas, izmantojot sekojošās pieejas:

1. tēzei - izveidota pētījumu atlases pieeja, identificēti, izpētīti un salīdzināti zināšanu izguves ietvari, noteikti to trūkumi papildinot zinātnisko literatūru par zināšanu izguves ietvariem;
2. tēzei - izstrādāta UIS-KDF ietvara pielietojuma kvantitatīvo rādītāju novērtējuma pieeja, UIS-KDF pielietojuma metodoloģija un izstrādāti UIS-KDF ietvara aprobācijas piemēri. UIS-KDF ietvars pielietots jaunuzņēmuma Monetization Solutions Inc. darbības nodrošināšanai un pētniecības projektu realizācijai, kas vērtēts, izmantojot izstrādāto kvantitatīvo rādītāju novērtējuma pieeju.

Darba praktiskā nozīme - izstrādāts adaptējams UIS-KDF ietvars, kas balstās mākoņskaitļošanas tehnoloģiju izmantošanā, ietvars pielietots jaunuzņēmuma organizācijas mērķu atbalstam tehnoloģisko procesu nodrošināšanai.

Galvenie praktiskie rezultāti:

- Izstrādātas trīs dažādu abstrakcijas līmeņu UIS-KDF aprobācijas;
- UIS-KDF ietvars pielietots jaunuzņēmuma darbības nodrošināšanai;

Galvenie teorētiskie rezultāti:

- Izstrādāta pētījumu atlases pieeja;
- Izstrādāta zināšanu izguves ietvaru novērtēšanas pieeja;
- Izstrādāta adaptējama UIS-KDF ietvara arhitektūra;
- Izstrādāta UIS-KDF ietvara pielietojuma kvantitatīvā novērtējuma pieeja;
- Izstrādāta UIS-KDF pielietošanas metodoloģija;
- Papildināta zinātniskā literatūra par zināšanu izguves ietvariem.

Galvenie Secinājumi:

1. Zināšanu izguve attiecas uz zināšanu izguves procesu datus un akcentē konkrētu datu izguves metožu augsta līmeņa pielietojumu. Zināšanas tiek iegūtas, izmantojot datu izguves algoritmus, lai saskaņā ar definētiem nosacījumiem iegūtu zināšanas, kā arī piemērotu nepieciešamās datu priekšapstrādes, apakšizlases un transformācijas.
2. Pasīvi uzkrātie sistēmas dati var nodrošināt efektīvāku un datus balstītu lēmumu pieņemšanas procesu, piemēram, hipotēžu validāciju, lai noteiktu kāda izskata komponentes darbojas labāk vai kādi kvalitātes kritēriji būtiski sistēmu lietotājiem.
3. Lietotāju reakcijas datu apjoms un sarežģītība lietojumprogrammās liecina, ka datu apstrādei ir nepieciešams nozīmīgs tehnisks darbs pie pārvaldības un infrastruktūras. Tādi zināšanu izguves procesu modeļi kā CRISP-DM datus iekļauj kā statistikas procesa vienības, kas nozīmē, ka zināšanu izguves procesu ir nepieciešams aplūkot kontekstā ar zināšanu izguves ietvaru, kurā process tiek pielietots.
4. Datu apstrādei nepieciešami precīzi definēti biznesa mērķi, darbībām jānodrošina pievienotā vērtība, ne tikai veikla apstrāde. Pievienoto vērtību no datiem iegūst tos analizējot un vizualizējot, kā arī izmantojot mašīnmācīšanās algoritmus, lai no datiem izgūtu lietderīgas zināšanas, kas atbalstītu nākotnes tendenču prognozēšanu un situāciju analīzi. Datu pakalpojumu patērētājiem pēc apstrādes un analīzes ir jāiegūst lietderīgas zināšanas, kas nozīmē, ka pieprasījums pēc zināšanu izguves sistēmām turpinās palielināties.
5. Zināšanu izguves sistēmas ir saistītas ar nepieciešamību uzkrāt pēc iespējas vairāk datu, lai pieņemtu datus balstītus lēmumus. Savukārt personu datu aizsardzības regula nosaka, ka noteiktiem un definētiem mērķiem jāizmanto tikai minimāli nepieciešamās datu kopas. Tas nozīmē, ka organizācijām ir jāizvērtē, vai konkrētas personas datu izmantošana atbilst attiecīgā datu subjekta saprātīgām sagaidāmajām prasībām, kas ir pretrunā ar liela apjoma datu uzkrāšanas sistēmu praksi.
6. Datu glabāšana, apstrāde, vizualizācija un mākoņskaitļošanas resursu pielietošana ir tikai daļa no zināšanu izguves sistēmu tehnoloģiskajiem apsvērumiem, procesā iesaistītas arī lietojumprogrammas un/vai lietu

interneta iekārtas. Tehnoloģiskie apsvērumi lietotnēm būtiski atšķirsies no lietojumprogrammas biznesa mērķiem. Zināšanu izguves sistēmām jāatbilst arī datu uzkrāšanas un glabāšanas regulējošajiem tiesiskajiem standartiem, kā arī tām jāiekļauj preventīvi un korektīvi risinājumi kiberaudraudējumu scenārijiem.

7. Kontroles sniegšana datu subjektam, kā to paredz VDAR, nozīmē, ka organizācijām jāņem vērā personu datu regulas nosacījumi. VDAR būtiski ietekmē zināšanu izguves sistēmas un to funkcionalitāti, ja vien dati netiek pilnībā anonimizēti.
8. Zināšanu izguves sistēmas elastības nodrošināšanai vēlams pielietot mākoņa bāzēto infrastruktūru. Uzturēt tradicionālu infrastruktūru un iekārtas, īpaši globālām sistēmām, ir ekonomisks izaicinājums. Neatbilstošas infrastruktūras izveide rada sarežģījumus gadījumos, kad zināšanu izguves sistēmai iteratīvi mainās prasības un esošā infrastruktūra neiekļauj jaunajām prasībām nepieciešamos resursus.
9. Zināšanu izguves sistēmas veido savstarpēji savienotas programmatūras sistēmas un zināšanu izguves sistēmu izstrādei pieejamo konceptuālo ietvaru skaits ir ierobežots un tie pielietojami kādā noteiktā sfērā, kas nav tiešā veidā pārnesami starp dažādām nozarēm.
10. UIS-KDF ietvara izstrādei būtisks nosacījums ir izmantot atbilstošu mākoņskaitļošanas pakalpojumu sniedzēju. Izveidot, uzturēt un nodrošināt atbilstošus risinājumus tikai ar izstrādātāja vai pasūtītāja resursiem ir potenciāli iespējams, tomēr šāda veida pieeja palielina tehnisko procesu sarežģītību.
11. UIS-KDF ietvars paredzēts zināšanu izguves sistēmu plānošanas agrai stadijai - infrastruktūras plānošanas atbalsta procesam par tehnoloģijām, to arhitektūru un plānošanu atbildīgajiem uzņēmumu darbiniekiem. UIS-KDF ietvara nolūks ir atvieglot zināšanu izguves sistēmu izstrādes sarežģīto un laikietilpīgo procesu, uzrādot arhitektūras plānošanas un organizācijas vadlīnijas.
12. Eksistējošos lielos uzņēmumos un organizācijās (>250 nodarbināto) UIS-KDF var nebūt piemērots, jo šīs organizācijas bieži vien ir ārpus zināšanu izguves sistēmu projektēšanas un plānošanas agrās stadijas, kurai paredzēts UIS-KDF. Šādiem uzņēmumiem jau ir izveidotas, nobriedušas

sistēmas un procesi, kas ir integrēti uzņēmumu esošajā IT infrastruktūrā un biznesa darbplūsmās.

13. UIS-KDF aprobācija jaunuzņēmuma darbības nodrošināšanai vienkāršoja uzņēmuma infrastruktūras izstrādes procesu, modulārais dalījums slāņos spēj nodrošināt atbilstošu globālu mērogojamību uzņēmumam paplašinot un diversificējot darbību. UIS-KDF pielietošana sniedza iespēju veikli sagatavot uzņēmumu SOC-2 sertifikācijas procesam un iekšējo procesu sagatavošanā VDAR atbilstībai.

IZMANTOTĀ LITERATŪRA

1. Abdi, H., Mohammadi-ivatloo, B., Javadi, S., Khodaei, A. R., & Dehnavi, E. (2017). Energy Storage Systems. In *Distributed Generation Systems* (pp. 333–368). Elsevier. <https://doi.org/10.1016/B978-0-12-804208-3.00007-8>
2. *Activity Monitor*. (2021). [Computer software]. BlindZone. <https://play.google.com/store/apps/details?id=com.ddm.deviceinfo>
3. *Anaconda*. (2022). [Computer software]. Continuum analytics. <https://www.anaconda.com/products/distribution>
4. Android Developers. (2020, March 27). *Processes and Application Lifecycle*. <https://developer.android.com/guide/components/activities/process-lifecycle>
5. Android Developers. (2021, July 17). *Profile battery usage with Batterystats and Battery Historian*. <https://developer.android.com/topic/performance/power/setup-battery-historian>
6. *Apache Flink*. (2021). [Computer software]. Apache Software Foundation. <https://flink.apache.org/>
7. *Apache Flume*. (2022). [Computer software]. Apache Software Foundation. <https://flume.apache.org/>
8. *Apache Hadoop*. (2021). [Computer software]. Apache Software Foundation. <https://hadoop.apache.org/>
9. *Apache Kafka*. (2022). [Computer software]. Apache Software Foundation. <https://kafka.apache.org/>
10. *Apache Mahout*. (2022). [Computer software]. Apache Software Foundation. <https://mahout.apache.org/>
11. *Apache Samza*. (2022). [Computer software]. Apache Software Foundation. <https://samza.apache.org/>
12. *Apache Sentry*. (2021). [Computer software]. Apache Software Foundation. <https://sentry.apache.org/>
13. *Apache Spark*. (2021). [Computer software]. Apache Software Foundation. <https://spark.apache.org/>
14. *Apache Sqoop*. (2022). [Computer software]. Apache Software Foundation. <https://sqoop.apache.org/>
15. *Apache Storm*. (2021). [Computer software]. Apache Software Foundation. <https://storm.apache.org/>

16. *Apple Pay*. (2022). [Computer software]. Apple Inc. <https://www.apple.com/apple-pay/>
17. Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., & Zaharia, M. (2010). A view of cloud computing. *Communications of the ACM*, 53(4), 50–58. <https://doi.org/10.1145/1721654.1721672>
18. *AWS Fargate*. (2022). [Computer software]. Amazon Inc. <https://aws.amazon.com/fargate/>
19. Baldi, P., Sadowski, P., & Lu, Z. (2017). Learning in the machine: The symmetries of the deep learning channel. *Neural Networks*, 95, 110–133. <https://doi.org/10.1016/j.neunet.2017.08.008>
20. Barthel, M., Mitchell, A., Asanre-Marfo, D., Kennedy, C., & Worden, K. (2020, August 12). *Measuring News Consumption in a Digital Era* [Research institute]. <https://www.pewresearch.org/journalism/2020/12/08/measuring-news-consumption-in-a-digital-era/>
21. Barzilay, M. (2019). *DATA COLLECTION AND MOBILE TECHNOLOGIES*. <https://doi.org/10.13140/RG.2.2.31541.93929>
22. Bernardo, J. M. (2011). Modern Bayesian Inference. In *Philosophy of Statistics* (pp. 263–306). Elsevier. <https://doi.org/10.1016/B978-0-444-51862-0.50008-3>
23. Berry, M. J. A., & Linoff, G. (1997). *Data mining techniques: For marketing, sales, and customer support*. Wiley.
24. Bigelow, S. J. (2021, July 21). 4 best practices for big data security in cloud computing. *TechTarget*. <https://www.techtarget.com/searchcloudcomputing/tip/4-best-practices-for-big-data-security-in-cloud-computing>
25. Bindé, J., Matsuura, K., & Unesco (Eds.). (2005). *Towards knowledge societies*. UNESCO Pub.
26. *Blackbox*. (2022). [Computer software]. Grow Pixel. <https://apps.apple.com/us/app/blackbox-a-clever-brain-game/id962969578>
27. Bok, K., Oh, H., Lim, J., Pae, Y., Choi, H., Lee, B., & Yoo, J. (2017). An efficient distributed caching for accessing small files in HDFS. *Cluster Computing*, 20(4), 3579–3592. <https://doi.org/10.1007/s10586-017-1147-2>
28. Bolisani, E., Borgo, S., & Oltramari, A. (2012). Using knowledge as an object: Challenges and implications. *Knowledge Management Research & Practice*, 10(3), 202–205. <https://doi.org/10.1057/kmrp.2012.32>
29. Bolisani, E., & Bratianu, C. (2018a). The Elusive Definition of Knowledge. In E.

- Bolisani & C. Bratianu, *Emergent Knowledge Strategies* (Vol. 4, pp. 1–22). Springer International Publishing. https://doi.org/10.1007/978-3-319-60657-6_1
30. Bolisani, E., & Bratianu, C. (2018b). The Elusive Definition of Knowledge. In E. Bolisani & C. Bratianu, *Emergent Knowledge Strategies* (Vol. 4, pp. 1–22). Springer International Publishing. https://doi.org/10.1007/978-3-319-60657-6_1
 31. Brownlee, J. (2020, March 7). *A Gentle Introduction to k-fold Cross-Validation*. <https://machinelearningmastery.com/k-fold-cross-validation/>
 32. Bunker, R. P., & Thabtah, F. (2019). A machine learning framework for sport result prediction. *Applied Computing and Informatics*, 15(1), 27–33. <https://doi.org/10.1016/j.aci.2017.09.005>
 33. Cabena, P. (Ed.). (1998). *Discovering data mining: From concept to implementation*. Prentice Hall.
 34. Candillier, L., Jack, K., Fessant, F., & Meyer, F. (2009). State-of-the-Art Recommender Systems: In M. Chevalier, C. Julien, & C. Soule-Dupuy (Eds.), *Collaborative and Social Information Retrieval and Access* (pp. 1–22). IGI Global. <https://doi.org/10.4018/978-1-60566-306-7.ch001>
 35. Carrol, A., & Heisner, G. (2010). An Analysis of Power Consumption in a Smartphone. *Proceedings of the 2010 USENIX Conference*. USENIX annual technical conference. https://www.researchgate.net/publication/215697458_An_Analysis_of_Power_Consumption_in_a_Smartphone
 36. Cavada, M., Hunt, D. V. L., & Rogers, C. D. F. (2014). *Smart Cities: Contradicting Definitions and Unclear Measures*. <https://doi.org/10.13140/2.1.1756.5120>
 37. Chapman Pete, Julian Clinton, Randy Kerber, Thomas Khabaza, Thomas Reinartz, Colin Shearer, & Rüdiger Wirth. (2000). *CRISP-DM 1.0: Step-by-step data mining guide*. <https://www.kde.cs.uni-kassel.de/wp-content/uploads/lehre/ws2012-13/kdd/files/CRISP-WP-0800.pdf>
 38. Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16, 321–357. <https://doi.org/10.1613/jair.953>
 39. Chen, L.-S., Lin, M.-R., & Pan, Y.-T. (2017). Find crucial factors of in-game purchase using neural networks. *2017 IEEE 8th International Conference on Awareness Science and Technology (iCAST)*, 334–338. <https://doi.org/10.1109/ICAwST.2017.8256473>
 40. Chen, M., Herrera, F., & Hwang, K. (2018). Cognitive Computing: Architecture,

- Technologies and Intelligent Applications. *IEEE Access*, 6, 19774–19783. <https://doi.org/10.1109/ACCESS.2018.2791469>
41. Cios, K. J., & Kurgan, L. A. (2005). Trends in Data Mining and Knowledge Discovery. In N. R. Pal & L. Jain (Eds.), *Advanced Techniques in Knowledge Discovery and Data Mining* (pp. 1–26). Springer London. https://doi.org/10.1007/1-84628-183-0_1
 42. *Containerd*. (2022). [Computer software]. The Linux Foundation. <https://containerd.io/docs/>
 43. *Cool Tool Pro*. (2022). [Computer software]. deviantstudio. <https://play.google.com/store/apps/details?id=ds.cooltool&hl=en&gl=US>
 44. Davenport, T. H., & Prusak, L. (1998). *Working knowledge: How organizations manage what they know*. Harvard Business School Press.
 45. Deagustini, C. A. D., Fulladoza Dalibón, S. E., Gottifredi, S., Falappa, M. A., Chesñevar, C. I., & Simari, G. R. (2013). Relational databases as a massive information source for defeasible argumentation. *Knowledge-Based Systems*, 51, 93–109. <https://doi.org/10.1016/j.knosys.2013.07.010>
 46. Developers Android. (2020, August 25). *Dumpsys*. <https://developer.android.com/studio/command-line/dumpsys>
 47. *Docker*. (2022). [Computer software]. Docker Inc. <https://www.docker.com/resources/what-container/>
 48. Dombrowski, E., Rotenberg, L., & Bick, M. (2013). *Theory of knowledge: Course companion* (2013 Edition). Oxford University Press.
 49. Douaioui, K., Fri, M., Mabroukki, C., & Semma, E. A. (2018). The interaction between industry 4.0 and smart logistics: Concepts and perspectives. *2018 International Colloquium on Logistics and Supply Chain Management (LOGISTIQUA)*, 128–132. <https://doi.org/10.1109/LOGISTIQUA.2018.8428300>
 50. Drake, E. (2015). *Preparing to Teach in an Active Learning Classroom: A Checklist for Success*. The Faculty Center for Innovative Teaching, Central Michigan University. https://my.methodistcollege.edu/ICS/icsfs/Preparing_to_teach_in_active_learning_classrooms.pdf?target=4b552ff8-82fe-44b6-9f2b-388c82fe1863
 51. Dugar, V. (2019, December 26). *Telemetry in Software*. <https://medium.com/justdevtalk/telemetry-in-software-7e2766a58cc0>
 52. Dunleavy, T. (2018). High School Algebra Students Busting the Myth about Mathematical Smartness: Counterstories to the Dominant Narrative “Get It Quick and Get It Right.” *Education Sciences*, 8(2), 58. <https://doi.org/10.3390/educsci8020058>

53. Edwin, N. M. (2014). Software Frameworks, Architectural and Design Patterns. *Journal of Software Engineering and Applications*, 07(08), 670–678. <https://doi.org/10.4236/jsea.2014.78061>
54. Eiropas Parlaments. (2021, March 25). Lielie dati: Definīcija, priekšrocības, sarežģījumi [Governmental]. *Lielie Dati*. <https://www.europarl.europa.eu/topics/lv/article/20210211STO97614/lielie-dati-definicija-prieksrocibas-sarezgijumi>
55. EIROPAS PARLAMENTA UN EIROPAS SAVIENĪBAS PADOME. (2016). *EIROPAS PARLAMENTA UN PADOMES REGULA (ES) 2016/679 par fizisku personu aizsardzību attiecībā uz personas datu apstrādi un šādu datu brīvu apriti un ar ko atceļ Direktīvu 95/46/EK (Vispārīgā datu aizsardzības regula)*. <https://eur-lex.europa.eu/legal-content/LV/TXT/HTML/?uri=CELEX:32016R0679&from=LV>
56. El Bouchefry, K., & de Souza, R. S. (2020). Learning in Big Data: Introduction to Machine Learning. In *Knowledge Discovery in Big Data from Astronomy and Earth Observation* (pp. 225–249). Elsevier. <https://doi.org/10.1016/B978-0-12-819154-5.00023-0>
57. Elsevier. (2022). *Scopus Tip & Trick: Search smarter, find faster*. <https://blog.scopus.com/posts/scopus-tip-trick-search-smarter-find-faster>
58. ETL VS ELT | DIFFERENCES AND USE CASES. (2021, May 4). *Data Valley*. <https://datavalley.technology/etl-vs-elt-differences-and-use-cases/>
59. *Exynos 8895*. (2017). [Computer software]. Samsung. <https://www.notebookcheck.net/Samsung-Exynos-8895-Octa-SoC-Benchmarks-and-Specs.210830.0.html>
60. Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). Knowledge discovery and data mining: Towards a unifying framework. *KDD'96: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, 82–88.
61. Ferreira, V. G., & Canedo, E. D. (2020). Design sprint in classroom: Exploring new active learning tools for project-based learning approach. *Journal of Ambient Intelligence and Humanized Computing*, 11(3), 1191–1212. <https://doi.org/10.1007/s12652-019-01285-3>
62. *FIFA Football*. (2021). [Computer software]. Electronic Arts. <https://play.google.com/store/apps/details?id=com.ea.gp.fifamobile>
63. Frias, Z., & Pérez Martínez, J. (2018). 5G networks: Will technology and policy

- collide? *Telecommunications Policy*, 42(8), 612–621.
<https://doi.org/10.1016/j.telpol.2017.06.003>
64. Garau, C. (2014). From Territory to Smartphone: Smart Fruition of Cultural Heritage for Dynamic Tourism Development. *Planning Practice & Research*, 29(3), 238–255.
<https://doi.org/10.1080/02697459.2014.929837>
 65. Gokalp, M. O., Kayabay, K., Akyol, M. A., Eren, P. E., & Kocyigit, A. (2016). Big Data for Industry 4.0: A Conceptual Framework. *2016 International Conference on Computational Science and Computational Intelligence (CSCI)*, 431–434.
<https://doi.org/10.1109/CSCI.2016.0088>
 66. Gómez, M., Rouvoy, R., Adams, B., & Seinturier, L. (2016). Mining test repositories for automatic detection of UI performance regressions in Android apps. *Proceedings of the 13th International Conference on Mining Software Repositories*, 13–24.
<https://doi.org/10.1145/2901739.2901747>
 67. Gooding, J., Crook, R., & Tomlin, A. S. (2015). Modelling of roof geometries from low-resolution LiDAR data for city-scale solar energy applications using a neighbouring buildings method. *Applied Energy*, 148, 93–104.
<https://doi.org/10.1016/j.apenergy.2015.03.013>
 68. *Google Cloud Run*. (2022). [Computer software]. Google Inc.
<https://cloud.google.com/run>
 69. *Google Pay*. (2022). [Computer software]. Google. <https://pay.google.com/>
 70. Google Play. (2021). *Top Charts*. <https://play.google.com/store/apps/top>
 71. Gsmarena. (2020). *Samsung Galaxy Note 20 Ultra 5G*.
https://www.gsmarena.com/samsung_galaxy_note20_ultra_5g-10261.php
 72. Han, B., & Windsor, J. (2013). An investigation of the smartphone user's in-game purchase intention. *International Journal of Mobile Communications*, 11(6), 617.
<https://doi.org/10.1504/IJMC.2013.057818>
 73. Hattie, J., & Timperley, H. (2007). The Power of Feedback. *Review of Educational Research*, 77(1), 81–112. <https://doi.org/10.3102/003465430298487>
 74. *HBase*. (2022). [Computer software]. Apache Software Foundation.
<https://hbase.apache.org/>
 75. Horowitz, D. (2021, February 2). *GPU Buying Guide: How To Choose the Right Graphics Card*. <https://www.hp.com/us-en/shop/tech-takes/gpu-buying-guide>
 76. IBM. (2005). *Analytics solutions unified method*.
<ftp://ftp.software.ibm.com/software/data/sw-library/services/ASUM.pdf>

77. IBM. (2022). *What is IT Infrastructure?* <https://www.ibm.com/topics/infrastructure>
78. IBM Inc. (2021a). *IaaS versus PaaS versus SaaS*. <https://www.ibm.com/cloud/learn/iaas-paas-saas>
79. IBM Inc. (2021b). *What is MapReduce?* [Official analytics documentation]. <https://www.ibm.com/topics/mapreduce>
80. *IBM Watson Explorer*. (2022). <https://www.ibm.com/docs/en/watson-explorer/12.0.x>
81. Jansevskis, M., & Osis, K. (2018). Machine Learning and on 5G Based Technologies Create New Opportunities to Gain Knowledge. *2018 2nd European Conference on Electrical Engineering and Computer Science (EECS)*, 376–381. <https://doi.org/10.1109/EECS.2018.00076>
82. Jansevskis, M., & Osis, K. (2020a). *FRAMEWORK AND KNOWLEDGE DISCOVERY IN LEARNING FEEDBACK DATA BASED ACTIVE LEARNING ECOSYSTEM FOR TEACHING SUPPORT*. 5700–5708. <https://doi.org/10.21125/iceri.2020.1226>
83. Jansevskis, M., & Osis, K. (2020b). *KNOWLEDGE DISCOVERY AND FRAMEWORK FOR PURCHASE BEHAVIOR ANALYSIS IN MOBILE GAMING APPLICATIONS*. *Iadis Digital Library, Single*, 247–251. ISBN: 978-989-8704-21-4
84. Jansevskis, M., & Osis, K. (2022). *STATE OF KNOWLEDGE DISCOVERY PROCESS MODELS AND FRAMEWORKS*. *SOCIETY. TECHNOLOGY. SOLUTIONS. Proceedings of the International Scientific Conference*, 2, 14. <https://doi.org/10.35363/ViA.sts.2022.81>
85. Jansevskis, M., & Osis, K. (2023). Knowledge Discovery Frameworks and Characteristics. *Baltic Journal of Modern Computing*, 11(4). <https://doi.org/10.22364/bjmc.2023.11.4.08>
86. Jansevskis, M., Osis, K. (2024). Securing the Future: The Role of Knowledge Discovery Frameworks. In: Sipola, T., Alatalo, J., Wolfmayr, M., Kokkonen, T. (eds) *Artificial Intelligence for Security*. Springer, Cham. https://doi.org/10.1007/978-3-031-57452-8_5
87. Jansevskis, M., & Osis, K. (2024b). User Interaction and Response-Based Knowledge Discovery Framework. In A. Lopata, D. Gudonienė, & R. Butkienė (Eds.), *Information and Software Technologies* (Vol. 1979, pp. 102–112). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-48981-5_8
88. Jeren, A. (2020, 1201). The impact of the GDPR on Big Data. *Tech GDPR*. <https://techgdpr.com/blog/impact-of-gdpr-on-big-data>
89. Jia, L. (2017). Machine Learning for Accurate Battery Run Time Prediction. *Technical*

- Disclosure Commons*. https://www.tdcommons.org/dpubs_series/878
90. Jia, Y. (2021). *Caffe* [Computer software]. UC Berkeley. <https://caffe.berkeleyvision.org/>
 91. Rollins J. B. (2015). *Foundational Methodology for Data Science*. IBM Analytics. <https://tdwi.org/~media/64511A895D86457E964174EDC5C4C7B1.PDF>
 92. Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245), 255–260. <https://doi.org/10.1126/science.aaa8415>
 93. Kaasila, J., Ferreira, D., Kostakos, V., & Ojala, T. (2012). Testdroid: Automated remote UI testing on Android. *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia - MUM '12*, 1. <https://doi.org/10.1145/2406367.2406402>
 94. Kang, J.-M., Sin-Seok, S., James, W., & Hong. (2011). Personalized Battery Lifetime Prediction for Mobile Devices based on Usage Patterns. *Journal of Computing Science and Engineering*, 5(4), 338–345. <https://doi.org/10.5626/JCSE.2011.5.4.338>
 95. Karunaratne, P., Karunasekera, S., & Harwood, A. (2017). Distributed stream clustering using micro-clusters on Apache Storm. *Journal of Parallel and Distributed Computing*, 108, 74–84. <https://doi.org/10.1016/j.jpdc.2016.06.004>
 96. Kastouni, M. Z., & Ait Lahcen, A. (2020). Big data analytics in telecommunications: Governance, architecture and use cases. *Journal of King Saud University - Computer and Information Sciences*, S131915782030553X. <https://doi.org/10.1016/j.jksuci.2020.11.024>
 97. Kechagia, M., Mitropoulos, D., & Spinellis, D. (2015). Charting the API minefield using software telemetry data. *Empirical Software Engineering*, 20(6), 1785–1830. <https://doi.org/10.1007/s10664-014-9343-7>
 98. *Kirin 970*. (2017). [Computer software]. HiSilicon, ARM Holdings, TSMC. <https://www.notebookcheck.net/HiSilicon-Kirin-970-SoC-Benchmarks-and-Specs.262356.0.html>
 99. *Knowledge Society*. (n.d.). International Encyclopedia of the Social Sciences. Retrieved May 25, 2022, from <https://www.encyclopedia.com/social-sciences/applied-and-social-sciences-magazines/knowledge-society>
 100. Kozik, R., Choraś, M., Ficco, M., & Palmieri, F. (2018). A scalable distributed machine learning approach for attack detection in edge computing environments. *Journal of Parallel and Distributed Computing*, 119, 18–26. <https://doi.org/10.1016/j.jpdc.2018.03.006>

101. Kristapsone, S. (2019). *Statistiskās analīzes metodes pētījumā* (1st ed.). Biznesa augstskola Turība.
102. *Kubernetes*. (2022). [Computer software]. The Linux Foundation. <https://kubernetes.io/docs/home/>
103. *Kudu*. (2022). [Computer software]. Apache Software Foundation. <https://kudu.apache.org/>
104. Lin Xudong, Guan Meijiao, Lin Kangqing, & Qiu Zhiwei. (2016). Analyzing the influencing factors of mobile game consumption. *2016 13th International Conference on Service Systems and Service Management (ICSSSM)*, 1–6. <https://doi.org/10.1109/ICSSSM.2016.7538489>
105. LMT. (2022). *Kas ir 4G?* <https://www.lmt.lv/lv/jautajumi-4g-internets>
106. Lombardo, C. T., McCarthy, B., Ryan, E., & Connors, M. (2018). *Product roadmaps relaunched*. O'Reilly.
107. Longbing Cao, Yanchang Zhao, Huaifeng Zhang, Dan Luo, Chengqi Zhang, & Park, E. K. (2010). Flexible Frameworks for Actionable Knowledge Discovery. *IEEE Transactions on Knowledge and Data Engineering*, 22(9), 1299–1312. <https://doi.org/10.1109/TKDE.2009.143>
108. López Martínez, P., Dintén, R., Drake, J. M., & Zorrilla, M. (2021). A big data-centric architecture metamodel for Industry 4.0. *Future Generation Computer Systems*, 125, 263–284. <https://doi.org/10.1016/j.future.2021.06.020>
109. Loshin, D. (2013). *Business intelligence: The savvy manager's guide* (2nd ed). Morgan Kaufmann.
110. Luger, G. F. (2005). *Artificial intelligence: Structures and strategies for complex problem solving* (5th ed). Addison-Wesley.
111. *LXD*. (2022). [Computer software]. Canonical Ltd. <https://linuxcontainers.org/lxd/introduction/>
112. Malheiro, A., Ribeiro, F., Leal Jamil, G., Rascao, J. P., & Mealha, O. (Eds.). (2018). *Handbook of Research on Knowledge Management for Contemporary Business Environments*. IGI Global. <https://doi.org/10.4018/978-1-5225-3725-0>
113. Mansouri, S. S., Karvelis, P., Georgoulas, G., & Nikolakopoulos, G. (2017). Remaining Useful Battery Life Prediction for UAVs based on Machine Learning * *This work has received partial funding from the European Union's Horizon 2020 Research and Innovation Programme under the Grant Agreement No.644128, AEROWORKS. *IFAC-PapersOnLine*, 50(1), 4727–4732. <https://doi.org/10.1016/j.ifacol.2017.08.863>

114. Marshall, J., & Mergenthaler, S. (2022, January 19). *These are the 3 ways knowledge can provide strategic advantage*. <https://www.weforum.org/agenda/2022/01/this-is-how-knowledge-can-bring-you-strategic-advance/>
115. Martínez-Plumed, F., Contreras-Ochando, L., Ferri, C., Flach, P., Hernández-Orallo, J., Kull, M., Lachiche, N., & Ramírez-Quintana, M. J. (2017). CASP-DM: Context Aware Standard Process for Data Mining. *arXiv:1709.09003 [Cs]*. <http://arxiv.org/abs/1709.09003>
116. Martínez-Plumed, F., Contreras-Ochando, L., Ferri, C., Hernández-Orallo, J., Kull, M., Lachiche, N., Ramírez-Quintana, M. J., & Flach, P. (2021). CRISP-DM Twenty Years Later: From Data Mining Processes to Data Science Trajectories. *IEEE Transactions on Knowledge and Data Engineering*, 33(8), 3048–3061. <https://doi.org/10.1109/TKDE.2019.2962680>
117. Mead, Nancy. (2006, January 30). SQUARE Process. *Cybersecurity & Infrastructure Security Agency*. <https://www.cisa.gov/uscert/bsi/articles/best-practices/requirements-engineering/square-process>
118. Mehmood, Y., Görg, C., Muehleisen, M., & Timm-Giel, A. (2015). Mobile M2M communication architectures, upcoming challenges, applications, and future directions. *EURASIP Journal on Wireless Communications and Networking*, 2015(1), 250. <https://doi.org/10.1186/s13638-015-0479-y>
119. Metri, G., Agrawal, A., Peri, R., & Weisong Shi. (2012). What is eating up battery life on my SmartPhone: A case study. *2012 International Conference on Energy Aware Computing*, 1–6. <https://doi.org/10.1109/ICEAC.2012.6471003>
120. Mohajeri, N., Assouline, D., Guiboud, B., Bill, A., Gudmundsson, A., & Scartezzini, J.-L. (2018). A city-scale roof shape classification using machine learning for solar energy applications. *Renewable Energy*, 121, 81–93. <https://doi.org/10.1016/j.renene.2017.12.096>
121. Mohsen, M., Nasarudin, F., Gani, A., Karim, A., Hashem, I. A. T., Siddiqa, A., & Yaqoob, I. (2017). Big IoT Data Analytics: Architecture, Opportunities, and Open Research Challenges. *IEEE Access*, 5, 5247–5261. <https://doi.org/10.1109/ACCESS.2017.2689040>
122. Monetization Solutions Inc. (2022). [Corporate]. <https://monetizr.com/>
123. MongoDB Inc. (2021). *What is NoSQL?* <https://www.mongodb.com/nosql-explained>

124. Moyle Steve & Jorge Alipio. (2001). *RAMSYS-A methodology for supporting rapid remote collaborative data mining projects*.
https://www.researchgate.net/publication/247329752_RAMSYS-A_methodology_for_supporting_rapid_remote_collaborative_data_mining_projects
125. *Multiclass and multioutput algorithms*. (2021).
<https://scikit-learn.org/stable/modules/multiclass.html>
126. *MXNet* (1.9.0). (2021). [Computer software]. Apache Software Foundation.
<https://mxnet.apache.org/>
127. Nam, T., & Pardo, T. A. (2011). Conceptualizing smart city with dimensions of technology, people, and institutions. *Proceedings of the 12th Annual International Digital Government Research Conference on Digital Government Innovation in Challenging Times - Dg.o '11*, 282. <https://doi.org/10.1145/2037556.2037602>
128. Naudet, Y., Guedria, W., & Chen, D. (2009). Systems Science for Enterprise Interoperability. *2009 International Conference on Interoperability for Enterprise Software and Applications China*, 107–113. <https://doi.org/10.1109/I-ESA.2009.57>
129. Neta, R., & Pritchard, D. (Eds.). (2009). *Arguing about knowledge*. Routledge.
130. Nonaka, I., & Takeuchi, H. (1995a). *The knowledge-creating company: How Japanese companies create the dynamics of innovation*. Oxford University Press.
131. Nonaka, I., & Takeuchi, H. (1995b). *The knowledge-creating company: How Japanese companies create the dynamics of innovation*. Oxford University Press.
132. *Oracle Berkeley DB*. (2022). [Computer software]. Oracle.
<https://www.oracle.com/database/technologies/related/berkeleydb.html>
133. *OrmLite*. (2022). [Computer software]. <https://ormlite.com/>
134. Osei-Bryson, K.-M., & Barclay, C. (Eds.). (2015). *Knowledge discovery process and methods to enhance organizational performance*. CRC Press, Taylor & Francis Group.
135. Osis, K., (2011). *Personāgo zināšanu pārvaldības sistēmu izstrāde* [PhD Thesis]. Rīgas Tehniskā Universitāte.
136. Osman, A. M. S. (2019). A novel big data analytics framework for smart cities. *Future Generation Computer Systems*, 91, 620–633.
<https://doi.org/10.1016/j.future.2018.06.046>
137. Oussous, A., Benjelloun, F.-Z., Ait Lahcen, A., & Belfkih, S. (2018). Big Data technologies: A survey. *Journal of King Saud University - Computer and Information Sciences*, 30(4), 431–448. <https://doi.org/10.1016/j.jksuci.2017.06.001>
138. Pantano, E., & Priporas, C.-V. (2016). The effect of mobile retailing on consumers'

- purchasing experiences: A dynamic perspective. *Computers in Human Behavior*, 61, 548–555. <https://doi.org/10.1016/j.chb.2016.03.071>
- 139.Parecki, A. (2022). *OAuth 2.0*. <https://oauth.net/2/>
- 140.Peppers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, 24(3), 45–77. <https://doi.org/10.2753/MIS0742-1222240302>
- 141.Peter Markie, & Edward N. Zalta. (2017, Fall). Rationalism vs. Empiricism. *Stanford Encyclopedia of Philosophy*. <https://plato.stanford.edu/archives/fall2017/entries/rationalism-empiricism/>
- 142.Phaal, R., Farrukh, C. J. P., & Probert, D. R. (2004). Technology roadmapping—A planning framework for evolution and revolution. *Technological Forecasting and Social Change*, 71(1–2), 5–26. [https://doi.org/10.1016/S0040-1625\(03\)00072-6](https://doi.org/10.1016/S0040-1625(03)00072-6)
- 143.Pramanik, P. K. D., Sinhababu, N., Mukherjee, B., Padmanaban, S., Maity, A., Upadhyaya, B. K., Holm-Nielsen, J. B., & Choudhury, P. (2019). Power Consumption Analysis, Measurement, Management, and Issues: A State-of-the-Art Review of Smartphone Battery and Energy Usage. *IEEE Access*, 7, 182113–182172. <https://doi.org/10.1109/ACCESS.2019.2958684>
- 144.*PyTorch* (1.10). (2021). [Computer software]. Facebook Inc. <https://pytorch.org/>
- 145.*Rancher*. (2022). [Computer software]. Rancher. <https://rancher.com/docs/>
- 146.*Realm Database*. (2022). [Computer software]. MongoDB. <https://realm.io/>
- 147.Rich, B. (2017, December 6). *NoSQL vs SQL — Which Database Type is Better For Big Data Applications*. <https://analyticsindiamag.com/nosql-vs-sql-database-type-better-big-data-applications>
- 148.Rietsche, R., Duss, K., Persch, J. M., & Sollner, M. (2018). Design and Evaluation of an IT-based Formative Feedback Tool to Foster Student Performance. *Conference: 39th International Conference on Information Systems (ICIS)*. https://www.researchgate.net/publication/329450233_Design_and_Evaluation_of_an_IT-based_Formative_Feedback_Tool_to_Foster_Student_Performance
- 149.Romero, M., Guédria, W., Panetto, H., & Barafort, B. (2020). Towards a Characterisation of Smart Systems: A Systematic Literature Review. *Computers in Industry*, 120, 103224. <https://doi.org/10.1016/j.compind.2020.103224>
- 150.Rosabeth Moss Kanter & Stanley S. Litow. (2009). *Informed and Interconnected: A Manifesto for Smarter Cities*. Harvard Business School Working Paper.

151. Rotondo, A., & Quilligan, F. (2020). Evolution Paths for Knowledge Discovery and Data Mining Process Models. *SN Computer Science*, 1(2), 109. <https://doi.org/10.1007/s42979-020-0117-6>
152. Sacha, D., Sedlmair, M., Zhang, L., Lee, J. A., Peltonen, J., Weiskopf, D., North, S. C., & Keim, D. A. (2017). What you see is what you can change: Human-centered machine learning by interactive visualization. *Neurocomputing*, 268, 164–175. <https://doi.org/10.1016/j.neucom.2017.01.105>
153. Sadler, D. R. (1989). Formative assessment and the design of instructional systems. *Instructional Science*, 18(2), 119–144. <https://doi.org/10.1007/BF00117714>
154. Salini, P., & Kanmani, S. (2013). Model Oriented Security Requirements Engineering (MOSRE) Framework for Web Applications. In N. Meghanathan, D. Nagamalai, & N. Chaki (Eds.), *Advances in Computing and Information Technology* (Vol. 177, pp. 341–353). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-31552-7_36
155. Sarpong, A. J. (2017). *Predictive Accuracy: A Misleading Performance Measure for Highly Imbalanced Data*. <https://support.sas.com/resources/papers/proceedings17/0942-2017.pdf>
156. Schatz, D., Bashroush, R., & Wall, J. (2017). Towards a More Representative Definition of Cyber Security. *The Journal of Digital Forensics, Security and Law*. <https://doi.org/10.15394/jdfsl.2017.1476>
157. Schwartz, D., & Te'eni, D. (Eds.). (2011). *Encyclopedia of Knowledge Management, Second Edition*: IGI Global. <https://doi.org/10.4018/978-1-59904-931-1>
158. *Scikit Learn*. (2022). [Computer software]. https://scikit-learn.org/stable/getting_started.html
159. scikit-learn developers. (2022). *Tuning the hyper-parameters of an estimator*. https://scikit-learn.org/stable/modules/grid_search.html#grid-search
160. Scopus. (2020, September). What is Field-weighted Citation Impact (FWCI)? *Scopus: Access and Use Support Center*. https://service.elsevier.com/app/answers/detail/a_id/14894/supporthub/scopus/~/~what-is-field-weighted-citation-impact-%28fwhci%29%3F/
161. Severtson R. B. (2021, December 15). *What is the Team Data Science Process?* <https://docs.microsoft.com/en-us/azure/architecture/data-science-process/lifecycle>
162. Shu, B., Chen, H., & Sun, M. (2017). Dynamic Load Balancing and Channel Strategy for Apache Flume Collecting Real-Time Data Stream. *2017 IEEE International Symposium on Parallel and Distributed Processing with Applications and 2017 IEEE*

- International Conference on Ubiquitous Computing and Communications (ISPA/IUCC)*, 542–549. <https://doi.org/10.1109/ISPA/IUCC.2017.00089>
163. Sifa, R., Hadiji, F., Runge, J., Drachen, A., Kersting, Cristian, & Bauckhage, C. (2015). Predicting Purchase Decisions in Mobile Free-to-Play Games. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 11(1), 79–85.
164. *Simple System Monitor*. (2021). [Computer software]. Darshan Parajuli. <https://play.google.com/store/apps/details?id=com.dp.sysmonitor.app>
165. Škoda, P., & Adam, F. (Eds.). (2020). *Knowledge discovery in big data from astronomy and earth observation: AstroGeoInformatics*. Elsevier.
166. *Slopes*. (2022). [Computer software]. Breakpoint Studio LLC. <https://apps.apple.com/us/app/slopes/id643351983>
167. *Slow rendering*. (2021, October 27). <https://developer.android.com/topic/performance/vitals/render>
168. *Snapdragon 845*. (2018). [Computer software]. Qualcomm. <https://www.notebookcheck.net/Qualcomm-Snapdragon-845-SoC-Benchmarks-and-Specs.299446.0.html>
169. *SpatiaLite*. (2022). [Computer software]. <https://www.gaia-gis.it/fossil/libspatialite/index>
170. *SQLite*. (2022). [Computer software]. <https://www.sqlite.org/>
171. *Stripe*. (2022). [Computer software]. Stripe Inc. <https://stripe.com/en-gb-lv>
172. *Subway Surfers*. (2021). [Computer software]. SYBO Games. <https://play.google.com/store/apps/details?id=com.kiloo.subwaysurf>
173. Synopsys. (2022). BSIMM. *Building Security in Maturity Model*. <https://www.bsimm.com/>
174. Technopedia Inc. (2017, August 18). *Knowledge Discovery*. <https://www.techopedia.com/definition/25827/knowledge-discovery-in-databases-kdd>
175. *TensorFlow*. (2022). [Computer software]. Google Inc. <https://www.tensorflow.org/>
176. Tozzi, C. (2020, May 6). *THE BENEFITS OF API-DRIVEN DEVELOPMENT AND HOW YOU CAN IMPLEMENT IT*. <https://www.mertech.com/blog/api-driven-development>
177. University of Leeds. (2022). *Literature searching explained*. https://library.leeds.ac.uk/info/1404/literature_searching/14/literature_searching_explained/4

178. University of Wollongong Australia. (2022). *How to search effectively*.
<https://uow.libguides.com/literaturereview/how>
179. Valsts valodas centrs, Latviešu valodas aģentūra, Kultūras informācijas centrs, & Latvijas zinātņu akadēmija. (2022a). *API* [Electronic dictionary].
<https://termini.gov.lv/atrast/API>
180. Valsts valodas centrs, Latviešu valodas aģentūra, Kultūras informācijas centrs, & Latvijas zinātņu akadēmija. (2022b). Virtuālā mašīna. In *Virtuālā mašīna*.
<https://termini.gov.lv/atrast/virtu%C4%81%C4%81%20ma%C5%A1%C4%ABna>
181. VARAM. (2017). *IKT arhitektūras vadlīnijas—IT risinājumu sadarbības specifificēšanas vadlīnijas*. Vides aizsardzības un reģionālās attīstības ministrija.
https://www.varam.gov.lv/sites/varam/files/content/files/pielikums_nr_2a.docx
182. Villanueva Zacarias, A. G., Reimann, P., & Mitschang, B. (2018). A framework to guide the selection and configuration of machine-learning-based data analytics solutions in manufacturing. *Procedia CIRP*, 72, 153–158.
<https://doi.org/10.1016/j.procir.2018.03.215>
183. Villinger, S. (2019, July 5). *What is RAM and Why is it Important?*
<https://www.avast.com/c-what-is-ram-memory>
184. *Walking Dead*. (2021). [Computer software]. Scopely.
<https://play.google.com/store/apps/details?id=com.scopely.headshot>
185. Wang, J., Yang, Y., Wang, T., Sherrat, R. S., & Zhang, J. (2020). Big data service architecture: A survey. *Journal of Internet Technology*, 21(2), 393–405.
<https://doi.org/10.3966/160792642020032102008>
186. Wei, J., He, J., Chen, K., Zhou, Y., & Tang, Z. (2017). Collaborative filtering and deep learning based recommendation system for cold start items. *Expert Systems with Applications*, 69, 29–39. <https://doi.org/10.1016/j.eswa.2016.09.040>
187. Wheeler, B. D. (2018). *Adopting Active Learning Classroom (ALC) Technology and Overcoming Barriers: A Faculty Development Intervention Model for Technology-Enhanced Learning Spaces* [University of Massachusetts Amherst].
<https://doi.org/10.7275/11881321.0>
188. White, M. J. (2022, January 24). *What is CPU usage, and how to fix high CPU usage*.
<https://www.digitaltrends.com/computing/what-is-cpu-usage-and-how-to-fix-it/>
189. Wiggins, B. L., Eddy, S. L., Wener-Fligner, L., Freisem, K., Grunspan, D. Z., Theobald, E. J., Timbrook, J., & Crowe, A. J. (2017). ASPECT: A Survey to Assess Student Perspective of Engagement in an Active-Learning Classroom. *CBE—Life Sciences*

- Education*, 16(2), ar32. <https://doi.org/10.1187/cbe.16-08-0244>
190. Wiginton, L. K., Nguyen, H. T., & Pearce, J. M. (2010). Quantifying rooftop solar photovoltaic potential for regional renewable energy policy. *Computers, Environment and Urban Systems*, 34(4), 345–357. <https://doi.org/10.1016/j.compenvurbsys.2010.01.001>
191. Wood, T. (2020). What is the F-score? *F-Score*. <https://deepai.org/machine-learning-glossary-and-terms/f-score>
192. Xin, Y., Kong, L., Liu, Z., Chen, Y., Li, Y., Zhu, H., Gao, M., Hou, H., & Wang, C. (2018). Machine Learning and Deep Learning Methods for Cybersecurity. *IEEE Access*, 6, 35365–35381. <https://doi.org/10.1109/ACCESS.2018.2836950>
193. Xindong Wu, Xingquan Zhu, Gong-Qing Wu, & Wei Ding. (2014). Data mining with big data. *IEEE Transactions on Knowledge and Data Engineering*, 26(1), 97–107. <https://doi.org/10.1109/TKDE.2013.109>
194. Xing, E. P., Ho, Q., Dai, W., Kim, J. K., Wei, J., Lee, S., Zheng, X., Xie, P., Kumar, A., & Yu, Y. (2015). Petuum: A New Platform for Distributed Machine Learning on Big Data. *IEEE Transactions on Big Data*, 1(2), 49–67. <https://doi.org/10.1109/TBDDATA.2015.2472014>
195. *Yarn*. (2022). [Computer software]. Facebook Inc. <https://yarnpkg.com/>
196. Zack, M. H. (1999). Developing a Knowledge Strategy. *California Management Review*, 41(3), 125–145. <https://doi.org/10.2307/41166000>
197. Zhai, Y., Ong, Y.-S., & Tsang, I. W. (2014). The Emerging “Big Dimensionality.” *IEEE Computational Intelligence Magazine*, 9(3), 14–26. <https://doi.org/10.1109/MCI.2014.2326099>
198. Zhang, Y., Wang, X., Liu, X., Liu, Y., Zhuang, L., & Zhao, F. (2013). Towards better CPU power management on multicore smartphones. *Proceedings of the Workshop on Power-Aware Computing and Systems - HotPower '13*, 1–5. <https://doi.org/10.1145/2525526.2525849>
199. Zhou, L., Pan, S., Wang, J., & Vasilakos, A. V. (2017). Machine learning on big data: Opportunities and challenges. *Neurocomputing*, 237, 350–361. <https://doi.org/10.1016/j.neucom.2017.01.026>

PIELIKUMI

I PIELIKUMS: Promocijas darba literatūras izpēte

P1 - Saturs

P1 - Uzdevumi

P1 - 1. Definētais meklēšanas scenārijs

P1 -2. Meklēšanas rezultāti

P1 - 2.1. Knowledge discovery: (Computer science category)

P1 - 2.1.1. Visvairāk citētās publikācijas ar atslēgvārdiem kopš 1991.gada:

P1 - 2.1.2. Visvairāk citētās publikācijas ar atslēgvārdiem kopš 2012.gada:

P1 - 2.1.3. Autori ar visvairāk publikācijām un viņu virziens (kopš 1991.gada):

P1 - 2.1.4. Autori ar visvairāk publikācijām un viņu virziens (kopš 2012.gada):

P1 - 2.1.5. Potenciāli saistošas visvairāk citētās publikācijas pēdējos piecos gados:

P1 - 2.2. Feedback data: (Computer science category)

P1 - 2.2.1. Visvairāk citētās publikācijas ar atslēgvārdiem kopš 1969.gada:

P1 - 2.2.2. Visvairāk citētās publikācijas ar atslēgvārdiem kopš 2012.gada:

P1 - 2.2.3. Autori ar visvairāk publikācijām un viņu virziens (kopš 1969.gada):

P1 - 2.2.4. Autori ar visvairāk publikācijām un viņu virziens (kopš 2012.gada):

P1 - 2.2.5. Potenciāli saistošas visvairāk citētās publikācijas pēdējos piecos gados:

P1 - 2.3. Smart systems: (Computer science category)

P1 - 2.3.1. Visvairāk citētās publikācijas ar atslēgvārdiem kopš 1979.gada:

P1 - 2.3.2. Visvairāk citētās publikācijas ar atslēgvārdiem kopš 2012.gada:

P1 - 2.3.3. Autori ar visvairāk publikācijām un viņu virziens (kopš 1979.gada):

P1 - 2.3.4. Autori ar visvairāk publikācijām un viņu virziens (kopš 2012.gada):

P1 - 2.3.5. Potenciāli saistošas visvairāk citētās publikācijas pēdējos piecos gados:

P1 - 3. Saistošās publikācijas

P1 - 3.1. Munir, K., & Sheraz Anjum, M. (2018). The use of ontologies for effective knowledge modelling and information retrieval.

P1 - 3.2. Osman, A. M. S. (2019). A novel big data analytics framework for smart cities.

P1 - 3.3. Cheng, X., Fang, L., Yang, L., & Cui, S. (2019). Mobile big data: The fuel for data-driven wireless.

P1 - 3.4. Romero, C., & Ventura, S. (2020). Educational data mining and learning analytics: An updated survey.

P1 - 3.5. Ristoski, P., & Paulheim, H. (2016). Semantic web in data mining and knowledge discovery: A comprehensive survey.

P1 - 3.6. Karpatne, A., Atluri, G., Faghmous, J. H., ..., Samatova, N., & Kumar, V. (2017). Theory-guided data science: A new paradigm for scientific discovery from data.

- P1 - 3.7. Bayer, I., He, X., Kanagal, B., & Rendle, S. (2017). A generic coordinate descent framework for learning from implicit feedback.
- P1 - 3.8. Yin, C., Ding, S., & Wang, J. (2019). Mobile marketing recommendation method based on user location feedback.
- P1 - 3.9. Gottipati, S., Shankararaman, V., & Gan, S. (2017). A conceptual framework for analyzing students' feedback.
- P1 - 3.10. Zhao, Q., Harper, F. M., Adomavicius, G., & Konstan, J. A. (2020). Explicit or implicit feedback? Engagement or satisfaction? A field experiment on machine-learning-based recommender systems.
- P1 - 3.11. Campos, A. R., Correia, A. T., Mourtzis, D., Margarito, A., & Ntalaperas, D. (2018). Engineering environment to support product-service design using value chain data.
- P1 - 3.12. Hatcher, W. G., & Yu, W. (2018). A survey of deep learning: Platforms, applications and emerging research trends.
- P1 - 3.13. Chen, M., Herrera, F., & Hwang, K. (2018). Cognitive computing: Architecture, technologies and intelligent applications.
- P1 - 3.14. Mabkhot, M. M., Al-Ahmari, A. M., Salah, B., & Alkhalefah, H. (2018). Requirements of the smart factory system: A survey and perspective.
- P1 - 3.15. Al-Turjman, F. (2019). 5G-enabled devices and smart-spaces in social-IoT: An overview.
- P1 - 3.16. Shafique, K., Khawaja, B. A., Sabir, F., Qazi, S., & Mustaqim, M. (2020). Internet of things (IoT) for next-generation smart systems: A review of current challenges, future trends and prospects for emerging 5G-IoT scenarios.
- P1 - 3.17. Peruzzini, M., & Pellicciari, M. (2017). A framework to design a human-centred adaptive manufacturing system for aging workers.
- P1 - 3.18. Kumar, N. M., & Mallick, P. K. (2018). The internet of things: Insights into the building blocks, component interactions, and architecture layers.
- P1 - 3.19. Ateya, A. A., Muthanna, A., Gudkova, I., ..., Vybornova, A., & Koucheryavy, A. (2018). Development of intelligent core network for tactile internet and future smart systems.
- P1 - 3.20. Eisele, S., Mardari, I., Dubey, A., & Karsai, G. (2017). RIAPS: Resilient information architecture platform for decentralized smart systems.
- P1 - 3.21. Al_Janabi, S. (2020). Smart system to create an optimal higher education environment using IDA and IOTs.
- P1 - 3.22. Elhoseny, H., Elhoseny, M., Riad, A. M., & Hassanien, A. E. (2018). A framework for big data analysis in smart cities.
- P1 - 3.23. Parra, L., Sendra, S., Lloret, J., & Rodrigues, J. J. P. C. (2017). Design and deployment of a smart system for data gathering in aquaculture tanks using wireless sensor networks.
- P1 - 3.24. Jacob, P. M., & Mani, P. (2018). Software architecture pattern selection model for internet of things based systems.
- P1 - 3.25. Bednar, P. M., & Welch, C. (2020). Socio-technical perspectives on smart working: Creating meaningful and sustainable systems.
- P1 - 3.26. Muhammad, G., Alhamid, M. F., & Long, X. (2019). Computing and processing on the edge: Smart pathology detection for connected healthcare.

P1 - 3.27. Zhang, J. a, Wang, Y. b, Li, S. b, & Shi, S. c. (2021). An architecture for IoT-enabled smart transportation security system: A geospatial approach.

P1 - 3.28. Rezultātu apkopojums

P1 - Uzdevumi

[1.1] - noteikt visvairāk citēto publikāciju:

- autorus;
- domēnus - problēmjomas;
- rīki priekš zināšanu izguves ko izmanto;
- kāda veida zināšamas tiek iegūtas.

[1.2] - izveidot pārskatu par iegūto informāciju.

[1.3] - rezultātu apkopojums:

- autora izvēlēta pētījuma sfēra;
- promocijas darba tematikai atbilstošās pētnieku publikācijas.

P1 - 1. Definētais meklēšanas scenārijs

Izvēlētais meklēšanas scenārijs nosaka vairāk citētās publikācijas, zināmākos un aktīvākos autorus. Scenārijs gan var neatklāt tiešām pominentos autorus, kuri nav pārāk aktīvi.

Meklējamie atslēgvārdi:

1. Knowledge discovery;
2. Feedback data;
3. Smart systems.

Pieci (5) meklēšanas scenāriji (top 10 rezultāti katrā):

1. Visvairāk citētās;
2. Visvairāk citētās pēdējos 10 gados;
3. Autori ar visvairāk publikācijām un viņu virziens (kopš 1991.gada);
4. Aktīvākie autori pēdējos 10 gados un viņu virziens;
5. Ar promocijas darba tematiku saistītās publikācijas no top 100 pēdējos piecos gados.

Sagaidāmais rezultāts: aptuveni 150 rezultātu kopa.

Atsevišķa literatūras izpēte: nākotnes uzdevumi pēdējos 5 gados katrā sfērā, visvairāk citētās.

P1 - 2. Meklēšanas rezultāti

P1 - 2.1. Knowledge discovery: (Computer science category)

Pirmās publikācijas parādās 1991.gadā (1 publikācija, Cai, Y., Cercone, N., & Han, J. (1991). Learning in relational databases: An attribute-oriented approach. (12 citāti))

P1 - 2.1.1. Visvairāk citētās publikācijas ar atslēgvārdiem kopš 1991.gada:

1. Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. (5262 cited)
2. He, H., & Garcia, E. A. (2009). Learning from imbalanced data. (3953 cited)
3. Kanungo, T., Mount, D. M., Netanyahu, N. S., ..., Silverman, R., & Wu, A. Y. (2002). An efficient k-means clustering algorithm: Analysis and implementation. (3370 cited)
4. Han, J., Kamber, M., & Pei, J. (2012). Data mining: Concepts and techniques. (3365 cited)
5. Bader, G. D., & Hogue, C. W. V. (2003). An automated method for finding molecular complexes in large protein interaction networks. (2713 cited)
6. Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). From data mining to knowledge discovery in databases. (2387 cited)
7. Pawlak, Z., & Skowron, A. (2007). Rudiments of rough sets. (1677 cited)
8. Tsoumakas, G., & Katakis, I. (2007). Multi-label classification: An overview. (1466 cited)
9. Chen, M.-S., Han, J., & Yu, P. S. (1996). Data mining: An overview from a database perspective.
10. Ludäscher, B., Altintas, I., Berkley, C., ..., Tao, J., & Zhao, Y. (2006). Scientific workflow management and the Kepler system.

P1 - 2.1.2. Visvairāk citētās publikācijas ar atslēgvārdiem kopš 2012.gada:

1. Han, J., Kamber, M., & Pei, J. (2012). Data mining: Concepts and techniques. (3365 cited)
2. Romero, C., & Ventura, S. (2013). Data mining in education. (411 cited)
3. Ge, Z., Song, Z., Ding, S. X., & Huang, B. (2017). Data mining and analytics in the process industry: The role of machine learning. (369 cited)
4. Dhar, V. (2013). Data science and prediction. (355 cited)
5. Kavakiotis, I., Tsave, O., Salifoglou, A., ..., Vlahavas, I., & Chouvarda, I. (2017). Machine learning and data mining methods in diabetes research. (347 cited)

6. Wu, Q., Ding, G., Xu, Y., ..., Wang, J., & Long, K. (2014). Cognitive internet of things: A new paradigm beyond connection. (312 cited)
7. Long, M., Wang, J., Ding, G., Pan, S. J., & Yu, P. S. (2014). Adaptation regularization: A general framework for transfer learning. (285 cited)
8. García, S., Luengo, J., Sáez, J. A., López, V., & Herrera, F. (2013). A survey of discretization techniques: Taxonomy and empirical analysis in supervised learning. (254 cited)
9. Li, J., Mei, C., & Lv, Y. (2013). Incomplete decision contexts: Approximate concept construction, rule acquisition and knowledge reduction. (196 cited)
10. Inbarani, H. H., Azar, A. T., & Jothi, G. (2014). Supervised hybrid feature selection based on PSO and rough sets for medical diagnosis. (195 cited)

P1 - 2.1.3. Autori ar visvairāk publikācijām un viņu virziens (kopš 1991.gada):

1. Holzinger, A. Biomedicine, bioinformatics, 50 documents.
2. Li, T. Computing approximations in set-valued systems, rough sets, rough approximations, 42 documents.
3. Napoli, A. Healthcare, ontology, bioinformatics, 39 documents.
4. Talia, D. Knowledge grid, data mining for large sets, distributed data mining, workflow programming, service-based middleware, cloud for scalable knowledge discovery, 38 documents.
5. Hong, W. Word sense disambiguation, attribute diagram, medicine, diabetes, 35 documents.
6. Cuzzocrea, A. Graph analytics, dense graph streams, multidimensional analysis of big data, knowledge from social graph data, big uncertain data, 34 documents.
7. Lavrač, N. Health care, literature mining, redescription mining, 34 publications.
8. Gibert, K. Environment, financial assets, 31 documents.
9. Han, J. Data mining, databases, spatial data warehouses, entity recognition, 30 publications.
10. Tsumoto, S. Rough sets, decision rules, 30 documents, not active since 2007.

P1 - 2.1.4. Autori ar visvairāk publikācijām un viņu virziens (kopš 2012.gada):

1. Holzinger, A. Biomedicine, bioinformatics, 50 documents.
2. Li, T. Computing approximations in set-valued systems, rough sets, rough approximations, 33 documents.

3. Hong, W. Word sense disambiguation, attribute diagram, medicine, diabetes, 31 documents.
4. Cuzzocrea, A. Graph analytics, dense graph streams, multidimensional analysis of big data, knowledge from social graph data, big uncertain data, 29 documents.
5. Leung, C. K. Dense graphs, knowledge from social graph data, knowledge from social networks, 29 documents (works with Cuzzocrea).
6. Napoli, A. Healthcare, ontology, bioinformatics, 25 documents.
7. Lavrač, N. Health care, literature mining, redescription mining, 20 documents.
8. Song, J. Biomedicine, medicine, 20 documents.
9. Li, S. Biomedicine, medicine, 18 documents (works with Hong, W., Song, J.).
10. Trovati, M. Big data extraction, artificial intuition, 17 documents.

P1 - 2.1.5. Potenciāli saistošas visvairāk citētās publikācijas pēdējos piecos gados:

1. Ge, Z., Song, Z., Ding, S. X., & Huang, B. (2017). Data mining and analytics in the process industry: The role of machine learning.
2. Karpatne, A., Atluri, G., Faghmous, J. H., ..., Samatova, N., & Kumar, V. (2017). Theory-guided data science: A new paradigm for scientific discovery from data.
3. Ramírez-Gallego, S., Krawczyk, B., García, S., Woźniak, M., & Herrera, F. (2017). A survey on data preprocessing for data stream mining: Current status and future directions.
4. Ibáñez, M.-B., & Delgado-Kloos, C. (2018). Augmented reality for STEM learning: A systematic review.
5. Bandaru, S., Ng, A. H. C., & Deb, K. (2017). Data mining methods for knowledge discovery in multi-objective optimization: Part A - Survey.
6. Munir, K., & Sheraz Anjum, M. (2018). The use of ontologies for effective knowledge modelling and information retrieval.
7. Osman, A. M. S. (2019). A novel big data analytics framework for smart cities.
8. Chen, M., Herrera, F., & Hwang, K. (2018). Cognitive computing: Architecture, technologies and intelligent applications.
9. Bujari, A., Ciman, M., Gaggi, O., & Palazzi, C. E. (2017). Using gamification to discover cultural heritage locations from geo-tagged photos.
10. García-Gil, D., Luengo, J., García, S., & Herrera, F. (2019). Enabling smart data: Noise filtering in big data classification.

11. Cheng, X., Fang, L., Yang, L., & Cui, S. (2019). Mobile big data: The fuel for data-driven wireless.
12. Romero, C., & Ventura, S. (2020). Educational data mining and learning analytics: An updated survey.
13. Luna, J. M., Castro, C., & Romero, C. (2017). MDM tool: A data mining framework integrated into Moodle.
14. Ristoski, P., & Paulheim, H. (2016). Semantic web in data mining and knowledge discovery: A comprehensive survey.

P1 - 2.2. Feedback data: (Computer science category)

Pirmie dokumenti parādās 1969 (1 dokuments) un tikai 1979 gadā pēc 10 gadu pārtraukuma atgriežas dokumenti par šo tēmu.

P1 - 2.2.1. Visvairāk citētās publikācijas ar atslēgvārdiem kopš 1969.gada:

1. Huang, Z., Chen, H., & Zeng, D. (2004). Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering. (473 cited)
2. Khan, S., & Yairi, T. (2018). A review on the application of deep learning in system health management. (306 cited)
3. Yu, X., Ren, X., Sun, Y., ..., Norick, B., & Han, J. (2014). Personalized entity recommendation: A heterogeneous information network approach. (284 cited)
4. Li, X., Cong, G., Li, X.-L., Pham, T.-A. N., & Krishnaswamy, S. (2015). Rank-geoFM: A ranking based geographical factorization method for point of interest recommendation. (221 cited)
5. Akdoğan, E., & Adli, M. A. (2011). The design and control of a therapeutic exercise robot for lower limb rehabilitation: Physiotherobot. (163 cited)
6. Shi, Y., Karatzoglou, A., Baltrunas, L., ..., Hanjalic, A., & Oliver, N. (2012). TFMAP: Optimizing MAP for top-n context-aware recommendation. (155 cited)
7. Miller, W. T. (1987). Sensor-based control of robotic manipulators using a general learning algorithm. (141 cited)
8. Hoi, S. C. H., Lyu, M. R., & Jin, R. (2006). A unified log-based relevance feedback scheme for image retrieval. (135 cited)
9. Lee, T. Q., Park, Y., & Park, Y.-T. (2008). A time-based approach to effective recommender systems using implicit feedback. (125 cited)

10. Choi, J., & Heath Jr., R. W. (2005). Interpolation based transmit beamforming for MIMO-OFDM with limited feedback. (112 cited)

P1 - 2.2.2. Visvairāk citētās publikācijas ar atslēgvārdiem kopš 2012.gada:

1. Khan, S., & Yairi, T. (2018). A review on the application of deep learning in system health management. (306 cited)
2. Yu, X., Ren, X., Sun, Y., ..., Norick, B., & Han, J. (2014). Personalized entity recommendation: A heterogeneous information network approach. (284 cited)
3. Li, X., Cong, G., Li, X.-L., Pham, T.-A. N., & Krishnaswamy, S. (2015). Rank-geoFM: A ranking based geographical factorization method for point of interest recommendation. (221 cited)
4. Shi, Y., Karatzoglou, A., Baltrunas, L., ..., Hanjalic, A., & Oliver, N. (2012). TFMAP: Optimizing MAP for top-n context-aware recommendation. (155 cited)
5. Bayer, I., He, X., Kanagal, B., & Rendle, S. (2017). A generic coordinate descent framework for learning from implicit feedback. (105 cited)
6. Yang, X., Steck, H., Guo, Y., & Liu, Y. (2012). On top-k recommendation using social networks. (103 cited)
7. He, X., Du, X., Wang, X., ..., Tang, J., & Chua, T.-S. (2018). Outer product-based neural collaborative filtering. (84 cited)
8. Karatzoglou, A., Baltrunas, L., Church, K., & Böhmer, M. (2012). Climbing the app wall: Enabling mobile app discovery through context-aware recommendations. (64 cited)
9. Ren, F., & Quan, C. (2012). Linguistic-based emotion analysis and recognition for measuring consumer satisfaction: An application of affective computing. (52 cited)
10. Lim, J.-S., Jang, W.-H., Yoon, G.-W., & Han, D.-S. (2013). Radio map update automation for WiFi positioning systems. (42 cited)

P1 - 2.2.3. Autori ar visvairāk publikācijām un viņu virziens (kopš 1969.gada):

1. Sawahashi, M., W-CDMA, last in 2005, 6 documents
2. Karatzoglou, A., Context aware representations, preference learning, mobile app discovery, 5 documents, latest from 2014
3. Baltrunas, L., Netflix, context aware recommendation, 4 documents, together with Karatzoglou, A, latest 2014
4. Chua, T.S., Recommender system, product based, latest in 2018, 4 documents

5. He, X., Recommender systems, learning from feedback data, visually aware recommender systems, latest 2021, 4 documents
6. Higuchi, K., W-CDMA, last from 2005, 4 documents
7. Jing, L., generative ranking for personalized recommendation, Deep Global and Local Generative Model for Recommendation, Deep Generative Recommendation with Maximizing Reciprocal Rank, 4 documents, latest 2020
8. Liu, H., The same as Jing, L, works together
9. Nower, N., Traffic pattern based data recovery scheme for cyber-physical systems, Efficient spatial data recovery scheme for cyber-physical system, Efficient temporal and spatial data recovery scheme for stochastic and incomplete feedback data of cyber-physical systems, Incomplete feedback data recovery scheme with Kalman filter for real-time cyber-physical systems, latest from 2015, 4 documents
10. Steck, H., On top-k recommendation using social networks, Gaussian ranking by matrix factorization, Multi-value probabilistic matrix factorization for IP-TV recommendations, Embarrassingly shallow autoencoders for sparse data, latest from 2019

P1 - 2.2.4. Autori ar visvairāk publikācijām un viņu virziens (kopš 2012.gada):

1. Karatzoglou, A., Context aware representations, preference learning, mobile app discovery, 5 documents, latest from 2014
2. Baltrunas, L., Netflix, context aware recommendation, 4 documents, together with Karatzoglou, A, latest 2014
3. He, X., Recommender systems, learning from feedback data, visually aware recommender systems, latest 2021, 4 documents
4. Jing, L., generative ranking for personalized recommendation, Deep Global and Local Generative Model for Recommendation, Deep Generative Recommendation with Maximizing Reciprocal Rank, 4 documents, latest 2020
5. Liu, H., The same as Jing, L, works together
6. Nower, N., Traffic pattern based data recovery scheme for cyber-physical systems, Efficient spatial data recovery scheme for cyber-physical system, Efficient temporal and spatial data recovery scheme for stochastic and incomplete feedback data of cyber-physical systems, Incomplete feedback data recovery scheme with Kalman filter for real-time cyber-physical systems, latest from 2015, 4 documents
7. Tan, Y. works together with Nower N, traffic pattern, cyber-physical, 4 documents

8. Yu, J., works together with Jing L, Liu H, recommendation systems 4 documents
9. Chen, R.S., Using cache optimization method to reduce network traffic in communication systems based on cloud computing, Design and Analysis of an Effective Two-Step Clustering Scheme to Optimize Prefetch Cache Technology, Cache Optimization Method to Reduce Network Traffic in Communication Systems, 3 documents, latest from 2019
10. Chen, Y.C., The same as Chen R.S, 3 documents

P1 - 2.2.5. Potenciāli saistošas visvairāk citētās publikācijas pēdējos piecos gados:

1. Bayer, I., He, X., Kanagal, B., & Rendle, S. (2017). A generic coordinate descent framework for learning from implicit feedback.
2. Khan, S., & Yairi, T. (2018). A review on the application of deep learning in system health management.
3. He, X., Du, X., Wang, X., ..., Tang, J., & Chua, T.-S. (2018). Outer product-based neural collaborative filtering.
4. Ding, J., Yu, G., He, X., ..., Jin, D., & Yu, J. (2018). Improving implicit recommender systems with view data.
5. Liu, H., Wu, Z., & Zhang, X. (2018). CPLR: Collaborative pairwise learning to rank for personalized recommendation.
6. Yin, C., Ding, S., & Wang, J. (2019). Mobile marketing recommendation method based on user location feedback.
7. Lin, C.-Y., Wang, L.-C., & Tsai, K.-H. (2018). Hybrid Real-Time Matrix Factorization for Implicit Feedback Recommendation Systems.
8. Gottipati, S., Shankararaman, V., & Gan, S. (2017). A conceptual framework for analyzing students' feedback.
9. Idrissi, N., & Zellou, A. (2020). A systematic literature review of sparsity issues in recommender systems.
10. Zhao, Q., Harper, F. M., Adomavicius, G., & Konstan, J. A. (2020). Explicit or implicit feedback? Engagement or satisfaction?: A field experiment on machine-learning-based recommender systems.
11. Saito, Y., Yaginuma, S., Nishino, Y., Sakata, H., & Nakata, K. (2020). Unbiased recommender learning from missing-not-at-random implicit feedback.

12. Campos, A. R., Correia, A. T., Mourtzis, D., Margarito, A., & Ntalaperas, D. (2018). Engineering environment to support product-service design using value chain data.
13. Wang, C., Zhou, T., Chen, C., Hu, T., & Chen, G. (Year). CAMO: A collaborative ranking method for content-based recommendation.

P1 - 2.3. Smart systems: (Computer science category)

1979 viens dokuments scopus, pēc tam 1983 viens, un pakāpeniski pieaugošs līdz 2019 gadam (385 dokumenti)

P1 - 2.3.1. Visvairāk citētās publikācijas ar atslēgvārdiem kopš 1979.gada:

1. Hossain, M. S., & Muhammad, G. (2016). Cloud-assisted Industrial Internet of Things (IIoT) - Enabled framework for health monitoring. (353 cited)
2. Wong, S. K., Ziarko, W., & Wong, P. C. N. (1985). Generalized vector space model in information retrieval. (217 cited)
3. Adika, C. O., & Wang, L. (2014). Autonomous appliance scheduling for household energy management. (182 cited)
4. Hatcher, W. G., & Yu, W. (2018). A Survey of Deep Learning: Platforms, Applications and Emerging Research Trends. (161 cited)
5. Li, Y., Dai, W., Ming, Z., & Qiu, M. (2016). Privacy Protection for Preventing Data Over-Collection in Smart City. (148 cited)
6. Cui, J., Xu, L., Bressler, S. L., Ding, M., & Liang, H. (2008). BSMART: A Matlab/C toolbox for analysis of multichannel neural time series. (139 cited)
7. Zhang, D.-G. (2012). A new approach and system for attentive mobile learning based on seamless migration. (138 cited)
8. Nayeri, R. D., Masri, S. F., Ghanem, R. G., & Nigbor, R. L. (2008). A novel approach for the structural identification and monitoring of a full-scale 17-story building based on ambient vibration measurements. (129 cited)
9. Varadan, V. K., & Varadan, V. V. (2000). Microsensors, microelectromechanical systems (MEMS), and electronics for smart structures and systems. (111 cited)
10. Al-Jaroodi, J., & Mohamed, N. (2019). Blockchain in Industries: A Survey. (106 cited)

P1 - 2.3.2. Visvairāk citētās publikācijas ar atslēgvārdiem kopš 2012.gada:

1. Hossain, M. S., & Muhammad, G. (2016). Cloud-assisted Industrial Internet of Things (IIoT) - Enabled framework for health monitoring. (353 cited)
2. Adika, C. O., & Wang, L. (2014). Autonomous appliance scheduling for household energy management. (182 cited)
3. Hatcher, W. G., & Yu, W. (2018). A Survey of Deep Learning: Platforms, Applications and Emerging Research Trends. (161 cited)
4. Li, Y., Dai, W., Ming, Z., & Qiu, M. (2016). Privacy Protection for Preventing Data Over-Collection in Smart City. (148 cited)
5. Zhang, D.-G. (2012). A new approach and system for attentive mobile learning based on seamless migration. (138 cited)
6. Al-Jaroodi, J., & Mohamed, N. (2019). Blockchain in Industries: A Survey. (106 cited)
7. Pramanik, M. I., Lau, R. Y. K., Demirkan, H., & Azad, M. A. K. (2017). Smart health: Big data enabled health paradigm within smart cities. (92 cited)
8. Goap, A., Sharma, D., Shukla, A. K., & Rama Krishna, C. (2018). An IoT based smart irrigation management system using Machine learning and open source technologies. (89 cited)
9. Choi, S.-B., Li, W., Yu, M., ..., Fu, J., & Do, P. X. (2016). State of the art of control schemes for smart systems featuring magneto-rheological materials. (85 cited)
10. Chamoso, P., González-Briones, A., Rodríguez, S., & Corchado, J. M. (2018). Tendencies of Technologies and Platforms in Smart Cities: A State-of-the-Art Review. (78 cited)

P1 - 2.3.3. Autori ar visvairāk publikācijām un viņu virziens (kopš 1979.gada):

1. Fummi, F., smart system design, 14 documents, active, interesting
2. Michel, B. smart systems, security, reliability, latest from 2018, 12 documents
3. Poncino, M. smart system design, Smart systems integration and simulation, Modeling and simulation of the power flow in smart systems, latest 2019, 12 documents
4. Gessner, T., smart system integration, latest 2016, 10 documents,
5. Lora, M., Automatic Generation of Analog/Mixed Signal Virtual Platforms for Smart Systems, smart system design, integration, planning, latest from 2020, 10 documents

6. Rodrigues, J.J.P.C., smart decision support systems, mobile systems, water demand forecasting, data accumulation cases, latest from 2020, 10 documents
7. Serdyukov, V.I., smart university designing, smart education, smart systems potential, latest from 2021, 10 documents
8. Serdyukova, N.A., works together with Serdyukov, V.I. 10 documents
9. Vinco, S., smart system design, methodology for simulation in smart system design, 10 documents, latest from 2019
10. Lloret, J., smart mobile systems, smart systems for data gathering, 9 documents, latest from 2019

P1 - 2.3.4. Autori ar visvairāk publikācijām un viņu virziens (kopš 2012.gada):

1. Fummi, F., smart system design, 14 documents, active, interesting
2. Poncino, M. smart system design, Smart systems integration and simulation, Modeling and simulation of the power flow in smart systems, latest 2019, 12 documents
3. Lora, M., Automatic Generation of Analog/Mixed Signal Virtual Platforms for Smart Systems, smart system design, integration, planning, latest from 2020, 10 documents
4. Rodrigues, J.J.P.C., smart decision support systems, mobile systems, water demand forecasting, data accumulation cases, latest from 2020, 10 documents
5. Serdyukov, V.I., smart university designing, smart education, smart systems potential, latest from 2021, 10 documents
6. Serdyukova, N.A., works together with Serdyukov, V.I. 10 documents
7. Vinco, S., smart system design, methodology for simulation in smart system design, 10 documents, latest from 2019
8. Lloret, J., smart mobile systems, smart systems for data gathering, 9 documents, latest from 2019
9. Sendra, S. smart mobile systems, works together with Rodrigues, J.J.P.C. and Lloret, J. 9 documents
10. Michel, B. smart systems, security, reliability, latest from 2018, 9 documents

P1 - 2.3.5. Potenciāli saistošas visvairāk citētās publikācijas pēdējos piecos gados:

1. Hatcher, W. G., & Yu, W. (2018). A Survey of Deep Learning: Platforms, Applications and Emerging Research Trends.

2. Pramanik, M. I., Lau, R. Y. K., Demirkan, H., & Azad, M. A. K. (2017). Smart health: Big data enabled health paradigm within smart cities.
3. Chamoso, P., González-Briones, A., Rodríguez, S., & Corchado, J. M. (2018). Tendencies of Technologies and Platforms in Smart Cities: A State-of-the-Art Review.
4. Chen, M., Herrera, F., & Hwang, K. (2018). Cognitive Computing: Architecture, Technologies and Intelligent Applications.
5. Mabkhot, M. M., Al-Ahmari, A. M., Salah, B., & Alkhalefah, H. (2018). Requirements of the smart factory system: A survey and perspective.
6. Al-Turjman, F. (2019). 5G-enabled devices and smart-spaces in social-IoT: An overview.
7. Shafique, K., Khawaja, B. A., Sabir, F., Qazi, S., & Mustaqim, M. (2020). Internet of things (IoT) for next-generation smart systems: A review of current challenges, future trends and prospects for emerging 5G-IoT Scenarios.
8. Peruzzini, M., & Pellicciari, M. (2017). A framework to design a human-centred adaptive manufacturing system for aging workers.
9. Kumar, N. M., & Mallick, P. K. (2018). The Internet of Things: Insights into the building blocks, component interactions, and architecture layers.
10. Ateya, A. A., Muthanna, A., Gudkova, I., ..., Vybornova, A., & Koucheryavy, A. (2018). Development of intelligent core network for tactile internet and future smart systems.
11. Eisele, S., Mardari, I., Dubey, A., & Karsai, G. (2017). RIAPS: Resilient information architecture platform for decentralized smart systems.
12. Al_Janabi, S. (2020). Smart system to create an optimal higher education environment using IDA and IOTs.
13. Elhoseny, H., Elhoseny, M., Riad, A. M., & Hassanien, A. E. (2018). A Framework for Big Data Analysis in Smart Cities.
14. Parra, L., Sendra, S., Lloret, J., & Rodrigues, J. J. P. C. (2017). Design and deployment of a smart system for data gathering in aquaculture tanks using wireless sensor networks.
15. Song, E. Y., Burns, M., & Pandey, A. (2019). IEEE 1451 Smart Sensor Digital Twin Federation for IoT/CPS Research.
16. Jacob, P. M., & Mani, P. (2018). Software architecture pattern selection model for internet of things based systems.

17. Bednar, P.M., Welch, C., Socio-Technical Perspectives on Smart Working: Creating Meaningful and Sustainable Systems, 2020
18. Muhammad, G., Alhamid, M.F., Long, X., Computing and processing on the edge: Smart pathology detection for connected healthcare, 2019

P1 - 3. Saistošās publikācijas

Uzdevumi:

- Kādi autori tiek citēti;
- Kādi ir nākotnes uzdevumi;
- Nākotnes tendences;
- Pētījuma sfēra;
- Pētījuma uzdevums;

Kategorizācijas vērtības:

- 1 - pētījumā tiek aplūkota cita sfēra, visticamāk promocijas darbam nebūs ieguvumu;
- 2 - daļēji atbilstoša sfēra un uzdevums;
- 3 - atbilstoša sfēra un uzdevumi, nepieciešama padziļināta izpēte;

P1 - 3.1 Munir, K., & Sheraz Anjum, M. (2018). The use of ontologies for effective knowledge modeling and information retrieval.

Nav tieši saistoša. Par to kā veidot datubāzu meklēšanas ontoloģijas. Optimizēt meklēšanas vaicājumus.

Score: 1

P1 - 3.2. Osman, A. M. S. (2019). A novel big data analytics framework for smart cities.

Paper presents a novel big data analytics framework for smart cities called “Smart City Data Analytics Panel — SCDAP”. Augsta reitinga publikācija. Tiek piedāvāts ietvars lielu datu apstrādei.

Nākotnes uzdevumi šīs publikācijas ietvaros:

- Develop efficient model persistence, retrieval and ensemble algorithms

- Develop efficient, powerful and friendly end-used interfaces involving multi-model visualization and OLAP style tools.
- Enable export/import of smart city analytical data models functionality to/from other analytical frameworks(e.g. knowledge exchange)
- Defining smart city performance measures (Key Performance Indicators — KPIs) in which analytics frameworks can be evaluated.

Prominentākais iekļautais literatūras avots: Chen, M., Mao, S., & Liu, Y. (2014). Big data: A survey.

Prominentie citētie:

- Tsai, C.-W., Lai, C.-F., Chao, H.-C., & Vasilakos, A. V. (2015). Big data analytics: A survey.
- Hashem, I. A. T., Chang, V., Anuar, N. B., Adewole, K., Yaqoob, I., Gani, A., Ahmed, E., & Chiroma, H. (2016). The role of big data in smart city.
- Al Nuaimi, E., Al Neyadi, H., Mohamed, N., & Al-Jaroodi, J. (2015). Applications of big data to smart cities.
- Chen, M., Mao, S., & Liu, Y. (2014). Big data: A survey.
- Singh, D., & Reddy, C. K. (2015). A survey on platforms for big data analytics.
- Santana, E. F. Z., Chaves, A. P., Gerosa, M. A., Kon, F., & Milojevic, D. S. (2017). Software platforms for smart cities: Concepts, requirements, challenges, and a unified reference architecture.
- Cooper, H. M. (1988). Organizing knowledge syntheses: A taxonomy of literature reviews.
- Strohbach, M., Ziekow, H., Gazis, V., & Akiva, N. (2015). Towards a big data analytics framework for IoT and smart city applications.
- Khan, Z., Anjum, A., Soomro, K., & Tahir, M. A. (2015). Towards cloud-based big data analytics for smart future cities.

Score: 3

P1 - 3.3. Cheng, X., Fang, L., Yang, L., & Cui, S. (2019). Mobile big data: The fuel for data-driven wireless.

In-depth and comprehensive coverage on the features, sources and applications of mobile big data, as well as the current state-of-the-art, challenges and opportunities for research and development in this field, with an emphasis on the user modeling, infrastructure supporting, data management, and knowledge discovery aspects.

Prominentākie citētie:

- González, M. C., Hidalgo, C. A., & Barabási, A.-L. (2008). Understanding individual human mobility patterns.
- Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning.
- Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets.

Score: 2

P1 - 3.4. Romero, C., & Ventura, S. (2020). Educational data mining and learning analytics: An updated survey.

This paper provides the current state of the art by reviewing the main publications, the key milestones, the knowledge discovery cycle, the main educational environments, the specific tools, the free available datasets, the most used methods, the main objectives, and the future trends in this research area.

Prominentākais citētais:

- Romero, C., & Ventura, S. (2010). Educational data mining: A review of the state of the art.

Score: 2

P1 - 3.5. Ristoski, P., & Paulheim, H. (2016). Semantic Web in data mining and knowledge discovery: A comprehensive survey

This survey article gives a comprehensive overview of Semantic Web data with the data mining and knowledge discovery approaches in different stages of the knowledge discovery process. As an example, they show how Linked Open Data can be used at various stages for building content-based recommender systems.

Prominentie autori:

- Bizer, C., Heath, T., & Berners-Lee, T. (2009). Linked data - The story so far.
- Gruber, T. R. (1995). Toward principles for the design of ontologies used for knowledge sharing.

Score: 1

P1 - 3.6. Karpatne, A., Atluri, G., Faghmous, J.H., Samatova, N., & Kumar, V. (2017). Theory-guided data science: A new paradigm for scientific discovery from data.

In this paper, we formally conceptualize the paradigm of TGDS (theory-guided data science) and present a taxonomy of research themes in TGDS. We describe several approaches for integrating domain knowledge in different research themes using illustrative examples from different disciplines. We also highlight some of the promising avenues of novel research for realizing the full potential of theory-guided data science.

Prominentie autori:

- Lazer, D., Kennedy, R., King, G., & Vespignani, A. (2014). The parable of Google flu: Traps in big data analysis.
- James, M. (2011). Big Data: The Next Frontier for Innovation, Competition, and Productivity.
- Ginsberg, J., Mohebbi, M.H., Patel, R.S., Brammer, L., Smolinski, M.S., & Brilliant, L. (2009). Detecting influenza epidemics using search engine query data.
- Friedman, J., Hastie, T., & Tibshirani, R. (2001). The Elements of Statistical Learning (1st ed.).

Score: 1

P1 - 3.7. Bayer, I., He, X., Kanagal, B., & Rendle, S. (2017). A generic coordinate descent framework for learning from implicit feedback.

In this paper, we provide a new framework for deriving efficient CD (coordinate descent) algorithms for complex recommender models. We identify and introduce the property of k-separable models. We show that k-separability is a sufficient property to

allow efficient optimization of implicit recommender problems with CD. We illustrate this framework on a variety of state-of-the-art models including factorization machines and Tucker decomposition. To summarize, our work provides the theory and building blocks to derive efficient implicit CD algorithms for complex recommender models.

Prominentie citētie:

- Rendle, S., Freudenthaler, C., & Schmidt-Thieme, L. (2010). Factorizing personalized Markov chains for next-basket recommendation.
- Yu, X., Ren, X., Sun, Y., Gu, Q., Sturt, B., Khandelwal, U., ... Han, J. (2014). Personalized entity recommendation: A heterogeneous information network approach.

Score: 2

P1 - 3.8. Yin, C., Ding, S., & Wang, J. (2019). Mobile marketing recommendation method based on user location feedback.

This paper focuses on location feedback data of user and proposes a location-based mobile marketing recommendation model by convolutional neural network (LBCNN). First, the users' location-based behaviors are divided into different time windows. For each window, the extractor achieves users' timing preference characteristics from different dimensions. Next, we use the convolutional model in the convolutional neural network model to train a classifier. The experimental results show that the model proposed in this paper is better than the traditional recommendation models in the terms of accuracy rate and recall rate, both of which increase nearly 10%

Score: 1

P1 - 3.9. Gottipati, S., Shankararaman, V., & Gan, S. (2017). A conceptual framework for analyzing students' feedback.

Capturing and analyzing the qualitative feedback data, at the individual course, school and university-level, can provide valuable insights on teaching practices and curriculum. In this paper, we propose a conceptual framework for student feedback analysis that provides the necessary structure for implementing a prototype tool for mining student comments. We then discuss the application of the tool to analyse feedback from selected courses.

Prominentie citētie:

- Biggs, J., & Tang, C. (2003). Teaching for Quality Learning at University. Society for Research into Higher Education and Open University Press.
- Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up? Sentiment classification using machine learning techniques.
- Esuli, A., & Sebastiani, F. (2006). SENTIWORDNET: A publicly available lexical resource for opinion mining.
- Hu, M., & Liu, B. (2004). Mining and summarizing customer reviews.

Score: 2

P1 - 3.10. Zhao, Q., Harper, F.M., Adomavicius, G., & Konstan, J.A. (2020). Explicit or implicit feedback? Engagement or satisfaction?: A field experiment on machine-learning-based recommender systems.

In this work, we evaluate six recommendation algorithms from a user-centric perspective, collecting both objective user activity data and subjective user perceptions. In a field experiment involving 1508 users who participated for at least a month, we compare six algorithms built using machine learning techniques, ranging from supervised matrix factorization, contextual bandit learning to Q learning. We found that the objective design in machine-learning-based recommender systems significantly affects user experience.

Score: 1

P1 - 3.11. Campos, A.R., Correia, A.T., Mourtzis, D., Margarito, A., & Ntalaperas, D. (2018). Engineering environment to support product-service design using value chain data.

This paper presents the early prototype of a software platform that supports companies in designing PSSs using feedback information collected over the value chain. The prototype described has been tested in three industrial companies and is currently being further developed.

Prominent citētie:

- Bitner, M. J., Ostrom, A. L., & Morgan, F. N. (2008). Service blueprinting: A practical technique for service innovation. *California Management Review, 50*(3), 66-94.

Score: 2

P1 - 3.12. Hatcher, W. G., & Yu, W. (2018). A survey of deep learning: Platforms, applications and emerging research trends.

In this paper, we seek to provide a thorough investigation of deep learning in its applications and mechanisms. Specifically, as a categorical collection of state of the art in deep learning research, we hope to provide a broad reference for those seeking a primer on deep learning and its various implementations, platforms, algorithms, and uses in a variety of smart-world systems.

Prominentie citētie:

- Agiwal, M., Roy, A., & Saxena, N. (2016). Next generation 5G wireless networks: A comprehensive survey.
- Buczak, A. L., & Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning.
- Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets.
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition.
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks.

Score: 2

P1 - 3.13. Chen, M., Herrera, F., & Hwang, K. (2018). Cognitive Computing: Architecture, Technologies and Intelligent Applications.

This paper proposes human-centered computing assisted by cognitive computing and cloud computing. First, we provide a comprehensive investigation of cognitive computing, including its evolution from knowledge discovery, cognitive science, and big data. Then, the system architecture of cognitive computing is proposed, which consists of

three critical technologies, i.e., networking (e.g., Internet of Things), analytics (e.g., reinforcement learning and deep learning), and cloud computing. Finally, it describes the representative applications of human-centered cognitive computing, including robot technology, emotional communication system, and medical cognitive system.

Prominentie citētie:

- Jin, J., Gubbi, J., Marusic, S., & Palaniswami, M. (2014). An information framework for creating a smart city through the internet of things.
- Catarinucci, L., De Donno, D., Mainetti, L., Palano, L., Patrono, L., Stefanizzi, M. L., & Tarricone, L. (2015). An IoT-Aware Architecture for Smart Healthcare Systems.
- Chen, M., Mao, S., & Liu, Y. (2014). Big data: A survey.
- Chen, M., Tian, Y., Fortino, G., Zhang, J., & Humar, I. (2018). Cognitive Internet of Vehicles.
- Fernández, A., del Río, S., López, V., Bawakid, A., del Jesus, M. J., Benítez, J. M., & Herrera, F. (Year). Big Data with Cloud Computing: An insight into the computing environment, MapReduce, and programming frameworks.
- Chen, M., Qian, Y., Hao, Y., Li, Y., & Song, J. (2018). Data-Driven Computing and Caching in 5G Networks: Architecture and Delay Analysis.
- Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., ... & Hassabis, D. (2016). Mastering the game of Go with deep neural networks and tree search.

Score 3

P1 - 3.14. Mabkhot, M. M., Al-Ahmari, A. M., Salah, B., & Alkhalefah, H. (2018). Requirements of the smart factory system: A survey and perspective.

The purpose of the current research is to highlight the perspectives that shape the smart factory and to suggest approaches and technical support to enable the realization of those perspectives. This paper fills this gap by identifying and analyzing research on smart factories. We suggest a framework to analyze existing research and investigate the elements and features of smart factory systems.

Prominent citătie:

- Schumacher, A., Erol, S., Sihh, W. (2016). A Maturity Model for Assessing Industry 4.0 Readiness and Maturity of Manufacturing Enterprises.
- Liu, X. F., Shahriar, M. R., Al Sunny, S. M. N., Leu, M. C., Hu, L. (2017). Cyber-physical manufacturing cloud: Architecture, virtualization, communication, and testbed.
- Lu, Y. (2017). Industry 4.0: A survey on technologies, applications and open research issues.
- Gokalp, M. O., Kayabay, K., Akyol, M. A., Eren, P. E., Kocyigit, A. (2016). Big data for Industry 4.0: A conceptual framework.
- Lee, J., Bagheri, B., Kao, H.-A. (2015). A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems.
- Qin, J., Liu, Y., Grosvenor, R. (2016). A Categorical Framework of Manufacturing for Industry 4.0 and beyond.
- Turner, C. J., Hutabarat, W., Oyekan, J., Tiwari, A. (2016). Discrete Event Simulation and Virtual Reality Use in Industry: New Opportunities and Future Trends.
- Wang, S., Wan, J., Li, D., Zhang, C. (2016). Implementing Smart Factory of Industrie 4.0: An Outlook.
- Wang, S., Wan, J., Zhang, D., Li, D., Zhang, C. (2016). Towards smart factory for industry 4.0: A self-organized multi-agent system with big data based feedback and coordination.
- Lin, Y.-C., Hung, M.-H., Huang, H.-C., Chen, C.-C., Yang, H.-C., Hsieh, Y.-S., Cheng, F.-T. (2017). Development of Advanced Manufacturing Cloud of Things (AMCoT)-A Smart Manufacturing Platform.

Score 3

P1 - 3.15. Al-Turjman, F. (2019). 5G-enabled devices and smart-spaces in social-IoT: An overview.

In this literature review, we present the main motivations in carrying these smart devices, and the correlation between the user surrounding context and the application usage. We focus on context-awareness in smart systems and space discovery paradigms;

online versus offline, the femtocell usage and energy aspects to be considered, and about the ongoing social IoT applications. Moreover, we highlight the most up-to-date open research issues in this area

Prominentie citētie

- Gupta, A., Jha, R.K. (2015). A Survey of 5G Network: Architecture and Emerging Technologies.
- Larsson, E.G., Edfors, O., Tufvesson, F., Marzetta, T.L. (2014). Massive MIMO for next generation wireless systems.
- Chen, S., Zhao, J. (2014). The requirements, challenges, and technologies for 5G of terrestrial mobile telecommunication.

Score 2

P1 - 3.16. Shafique, K., Khawaja, B.A., Sabir, F., Qazi, S., Mustaqim, M. (2020). Internet of Things (IoT) for next-generation smart systems: A review of current challenges, future trends and prospects for emerging 5G-IoT Scenarios.

This paper presents the IoT technology from a bird's eye view covering its statistical/architectural trends, use cases, challenges and future prospects. The paper also presents a detailed and extensive overview of the emerging 5G-IoT scenario. Fifth Generation (5G) cellular networks provide key enabling technologies for ubiquitous deployment of the IoT technology.

Prominentie citētie:

- Atzori, L., Iera, A., Morabito, G. (2010). The Internet of Things: A survey.
- Osseiran, A., Boccardi, F., Braun, V., Kusume, K., Marsch, P., Maternia, M., Queseth, O., ... Fallgren, M. (2014). Scenarios for 5G mobile and wireless communications: The vision of the METIS project.
- Yaqoob, I., Ahmed, E., Hashem, I.A.T., Ahmed, A.I.A., Gani, A., Imran, M., Guizani, M. (2017). Internet of Things Architecture: Recent Advances, Taxonomy, Requirements, and Open Challenges.
- Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions.

- Kaur, N., Sood, S.K. (2017). An Energy-Efficient Architecture for the Internet of Things (IoT).
- Zhou, J., Leppanen, T., Harjula, E., Ylianttila, M., Ojala, T., Yu, C., Jin, H. (2013). CloudThings: A common architecture for integrating the Internet of Things with Cloud Computing.
- Cirani, S., Davoli, L., Ferrari, G., Leone, R., Medagliani, P., Picone, M., Veltri, L. (2014). A scalable and self-configuring architecture for service discovery in the internet of things.
- Lee, I., Lee, K. (2015). The Internet of Things (IoT): Applications, investments, and challenges for enterprises.
- Lloret, J., Tomas, J., Canovas, A., Parra, L. (2016). An Integrated IoT Architecture for Smart Metering.
- Zanella, A., Bui, N., Castellani, A., Vangelista, L., Zorzi, M. (2014). Internet of things for smart cities.
- Gupta, A., Jha, R.K. (2015). A Survey of 5G Network: Architecture and Emerging Technologies.

Score: 3

P1 - 3.17. Peruzzini, M., Pellicciari, M. (2017). A framework to design a human-centered adaptive manufacturing system for aging workers.

This research focuses on the design of human-centered adaptive manufacturing systems (AMS) for modern companies, where aging workers are more and more common. In particular, it defines a methodology to design AMS able to adapt to the aging workers' needs considering their reduced workability, due to both physical and cognitive functional decrease, with the final aim to improve the human-machine interaction and the workers' wellbeing.

Score: 1

P1 - 3.18. Kumar, N.M., Mallick, P.K. (2018). The Internet of Things: Insights into the building blocks, component interactions, and architecture layers.

This paper deals with the internet of things (IoT) which has become a promising and vibrant technology to build powerful smart systems to monitor and analyze various

real time operating systems. In recent years a wide range of IoT applications have been developed. To understand the IoT concept, this paper studies the insights into the four building blocks of IoT (Things, Gateways, Network infrastructure, and Cloud infrastructure), three main components of IoT (The Things with Networked Sensors and Actuators, Raw Information and Processed Data Stores, and Analytical and Computing Engines) along with architecture layers (Three Layer, Five Layer, Six Layer, Seven Layer, Cloud, and FOG). The interaction between three components of IoT is also presented. The main contribution of this paper is that it summarizes the IoT, IoT building blocks, components and their interactions along with architecture layers systematically

Prominentie citētie:

- Bandyopadhyay, D., Sen, J. (2011). Internet of things: Applications and challenges in technology and standardization.
- Ray, P.P. (2018). A survey on Internet of Things architectures.
- Atzori, L., Iera, A., Morabito, G. (2010). The Internet of Things: A survey.
- Zanella, A., Bui, N., Castellani, A., Vangelista, L., Zorzi, M. (2014). Internet of things for smart cities.
- Xu, L.D., He, W., Li, S. (2014). Internet of things in industries: A survey.
- Sethi, P., Sarangi, S.R. (2017). Internet of Things: Architectures, Protocols, and Applications.
- Khan, R., Khan, S.U., Zaheer, R., Khan, S. (2012). Future internet: The internet of things architecture, possible applications and key challenges.
- Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions.
- Bonomi, F., Milito, R., Natarajan, P., Zhu, J. (2014). Fog computing: A platform for internet of things and analytics.
- Bonomi, F., Milito, R., Zhu, J., Addepalli, S. (2012). Fog computing and its role in the internet of things.

Score: 3

P1 - 3.19. Ateya, A.A., Muthanna, A., Gudkova, I., Vybornova, A., Koucheryavy, A. (2018). Development of intelligent core network for tactile internet and future smart systems.

This paper introduces a Tactile Internet system structure, which employs SDN in the core of the cellular network and mobile edge computing (MEC) in multi-levels. The work is mainly concerned with the structure of the core network. The system is simulated over a reliable environment and introduces a round trip latency of orders of 1 ms.

Score: 1

P1 - 3.20. Eisele, S., Mardari, I., Dubey, A., Karsai, G. (2017). RIAPS: Resilient information architecture platform for decentralized smart systems.

In this paper, we briefly describe a component-based decentralized software platform called Resilient Information Architecture Platform for Smart Systems (RIAPS) which provides an infrastructure for such systems. We briefly describe some initial applications built using this platform. Then, we focus on the design and integration choices for a resilient Discovery Manager service that is a critical component of this infrastructure. The service allows applications to discover each other, work collaboratively, and ensure the stability of the Smart System.

Prominent citătie:

- Lopez, P.G., Montresor, A., Epema, D., Datta, A., Higashino, T., Iamnitchi, A., Barcellos, M., (...), Riviere, E. (2015). Edge-centric computing: Vision and challenges.
- Stubbs, J., Moreira, W., Dooley, R. (2015). Distributed Systems of Microservices Using Docker and Serfnode.
- Cirani, S., Davoli, L., Ferrari, G., Leone, R., Medagliani, P., Picone, M., Veltri, L. (2014). A scalable and self-configuring architecture for service discovery in the internet of things.

Score: 3

P1 - 3.21. Al_Janabi, S. (2020). Smart system to create an optimal higher education environment using IDA and IoTs.

The designed system described in this paper consists of five stages to solve one of the higher education environments using intelligent data analysis (IDA) and the Internet of things (IOTs). The main points of this system are (i) it reduces the time rate and effort

wasted in taking daily attendance of students in each lecture. (ii) Increased accuracy in the administration reduces the need for human effort (i.e. the traditional way is the college university used a group of employees to enter the information manually for these systems, which increases the likelihood of errors). (iii) Increase the efficiency of teachers themselves because the lecture time now is set automatically from the moment he/she enters the lecture hall to the time he/she exits from it. (v) This integrated system is considered part of the e-government desired application in Iraq.

Score: 1

P1 - 3.22. Elhoseny, H., Elhoseny, M., Riad, A.M., & Hassanien, A.E. (2018). A Framework for Big Data Analysis in Smart Cities.

Due to the rapid change in technologies, new data forms exist which lead to a huge data size on the internet. As a result, some learning platforms such as e-learning systems must change their methodologies for data processing to be smarter. This paper proposes a framework for smoothly adapt the traditional e-learning systems to be suitable for smart cities applications. Learning Analytics (LA) has turned into a noticeable worldview with regards to instruction of late which embraces the current progressions of innovation, for example, cloud computing, big data processing, and Internet of Things. LA additionally requires a concentrated measure of preparing assets to create applicable investigative outcomes. Be that as it may, the customary methodologies have been wasteful at handling LA difficulties.

Prominentie citētie:

- Harshawardhan, S., & Devendra, P. (2014). A review paper on big data and Hadoop.
- Siemens, G., & Long, P. (2011). Penetrating the fog: Analytics in learning and education.

Score: 2

P1 - 3.23. Parra, L., Sendra, S., Lloret, J., & Rodrigues, J.J.P.C. (2017). Design and deployment of a smart system for data gathering in aquaculture tanks using wireless sensor networks.

This paper presents the design, deployment, and test of a smart data gathering system for monitoring several parameters in aquaculture tanks using a wireless sensor network. The system based on a server is able to request and collect data from several nodes and store them in a database. This information can be postprocessed to take efficient decisions. The paper also presents the design of a conductivity sensor and a level sensor. These sensors are installed in several aquaculture tanks. The system was implemented using Flyport modules. Finally, the data gathering system was tested in terms of consumed bandwidth and the delay Transmission Control Protocol (TCP) packets delivering data from the sensors.

Prominent citētie:

- Yick, J., Mukherjee, B., & Ghosal, D. (2008). Wireless sensor network survey.
- Wang, N., Zhang, N., & Wang, M. (2006). Wireless sensors in agriculture and food industry - Recent development and future perspective.
- Luo, C., Wu, F., Sun, J., & Chen, C.W. (2009). Compressive data gathering for large-scale wireless sensor networks.

Score: 2

P1 - 3.24. Jacob, P.M., & Mani, P. (2018). Software architecture pattern selection model for Internet of Things based systems.

Though various IoT developers follow their own design strategies for IoT architecture, there is a need for domain specific architecture for IoT applications. The authors propose a software architecture pattern selection model which can be followed in designing IoT systems. Non-functional requirements such as scalability, availability, reliability, security and heterogeneity determine the decision making of the proposed model. Analytical results prove that this model can be used as a reference model, for IoT developers in choosing a suitable software architecture pattern for their IoT system.

Prominent citētie:

- Bahga, A., & Madiseti, V. (2015). Internet of Things: A Hands-on Approach.

- Shaw, M., & Garlan, D. (1996). Software Architecture: Perspectives on An Emerging Discipline.
- Mall, R. (2014). Fundamentals of Software Engineering.
- Jansen, A., & Bosch, J. (2005). Software architecture as a set of architectural design decisions.
- Pressman, R. S. (2005). Software Engineering – A Practitioner's Approach.
- Sommerville, I. (2017). Software Engineering.

Score: 3

P1 - 3.25. Bednar, P. M., & Welch, C. (2020). Socio-Technical Perspectives on Smart Working: Creating Meaningful and Sustainable Systems.

This paper examines these developments from the perspective of unique, individual understandings of work roles and sustainability, posing the questions ‘Smart’ from whose point of view? Do smart systems promote sustainable organizations? How should design of smart systems be approached? It suggests that contemporary socio-technical systems approaches to organizational analysis are the best way to support harnessing of smart technologies in organizations.

Prominent citētie:

- Bateson, G. (1972). Steps to an Ecology of Mind.
- Baumeister, R. F., & Leary, M. R. (1995). The Need to Belong: Desire for Interpersonal Attachments as a Fundamental Human Motivation.
- Argyris, C., & Schön, D. A. (1974). Theory in Practice: Increasing Professional Effectiveness.
- Berger, P. L., & Luckmann, T. (1967). The Social Construction of Reality: A Treatise in the Sociology of Knowledge.
- Bertalanffy, L. V. (1969). General System Theory.
- Braverman, H. (1974). Labor and Monopoly Capital: The Degradation of Work in the Twentieth Century.
- Checkland, P. B. (1999). Systems Thinking, Systems Practice: A 30-Year Retrospective.
- Chesbrough, H. (2003). Open Innovation: The New Imperative for Creating and Profiting from Technology.

- Davenport, L., & Prusak, L. (2000). Working Knowledge: How Organizations Manage What They Know.
- Lee, J., Bagheri, B., & Kao, H.-A. (2015). A Cyber-Physical Systems Architecture for Industry 4.0-based Manufacturing Systems.

Score: 3

P1 - Muhammad, G., Alhamid, M. F., & Long, X. (2019). Computing and processing on the edge: Smart pathology detection for connected healthcare.

In this article, we propose a new smart pathology detection system using deep learning, edge computing, and cloud computing. Sensors will capture electroencephalogram (EEG) signals of a person and send the signals to a nearby edge computing server. The server will distribute a preprocessing step to available edge devices. The preprocessed signal will then be sent to a cloud computing server. In the cloud server, a proposed tree-based deep model will extract deep features from the EEG signal. The classified decision of whether the signal belongs to a normal person or a pathological person will be distributed to the stakeholders.

Prominent citētie:

- Chen, M., Li, W., Hao, Y., Qian, Y., & Humar, I. (2018). Edge cognitive computing based smart healthcare system.
- Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge Computing: Vision and Challenges.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions.

Score: 3

P1 - 3.27. Zhang, J., Wang, Y., Li, S., & Shi, S. (2021). An Architecture for IoT-Enabled Smart Transportation Security System: A Geospatial Approach.

This article aims to apply a geospatial modeling approach to propose smart transportation security systems (STSSs). It is modeled and simulated by undertaking an

experimental study in the city of Beijing, China. The simulation outcome of the proposed architecture is expected to offer a strategic guide for strategic security management of urban smart transportation.

Prominent citētie:

- Lin, J., Yu, W., Zhang, N., Yang, X., Zhang, H., & Zhao, W. (2017). A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications.
- Yang, Y., Wu, L., Yin, G., Li, L., & Zhao, H. (2017). A Survey on Security and Privacy Issues in Internet-of-Things.
- Rathore, M. M., Ahmad, A., Paul, A., & Rho, S. (2016). Urban planning and building smart cities based on the Internet of Things using Big Data analytics.

Score: 2

P1 - 3.28. Rezultātu apkopojums

Pētījumi, kurus nepieciešams izmantot tālākai analīzei. Apskatot kādus ietvarus piedāvā autori, gan kādus ietvarus piedāvā tie autori, kurus citē un izmanto publikāciju veidotāji.

Sarakstā iekļautas tās publikācijas, kuru vērtība ir 3.

P1 - 3.1. tabula. Rezultātu apkopojums (autora veidota)

Publikācija	Saistītie, potenciāli lietderīgie pētījumi
3.2.Osman, A.M.S. (2019). A novel big data analytics framework for smart cities.	<ul style="list-style-type: none"> • Tsai, C.-W., Lai, C.-F., Chao, H.-C., & Vasilakos, A.V. (2015). Big data analytics: a survey. • Hashem, I.A.T., Chang, V., Anuar, N.B., Adewole, K., Yaqoob, I., Gani, A., Ahmed, E., ... Chiroma, H. (2016). The role of big data in smart city. • Al Nuaimi, E., Al Neyadi, H., Mohamed, N., & Al-Jaroodi, J. (2015). Applications of big data to smart cities.

	<ul style="list-style-type: none"> ● Chen, M., Mao, S., & Liu, Y. (2014). Big data: A survey. ● Singh, D., & Reddy, C.K. (2015). A survey on platforms for big data analytics. ● Santana, E.F.Z., Chaves, A.P., Gerosa, M.A., Kon, F., & Milojicic, D.S. (2017). Software platforms for smart cities: Concepts, requirements, challenges, and a unified reference architecture. ● Cooper, H.M. (1988). Organizing knowledge syntheses: A taxonomy of literature reviews. ● Strohbach, M., Ziekow, H., Gazis, V., & Akiva, N. (2015). Towards a Big Data Analytics Framework for IoT and Smart City Applications. ● Khan, Z., Anjum, A., Soomro, K., & Tahir, M.A. (2015). Towards cloud based big data analytics for smart future cities.
<p>3.13.Chen, M., Herrera, F., & Hwang, K. (2018). Cognitive Computing: Architecture, Technologies and Intelligent Applications.</p>	<ul style="list-style-type: none"> ● Jin, J., Gubbi, J., Marusic, S., & Palaniswami, M. (2014). An information framework for creating a smart city through Internet of Things. ● Catarinucci, L., De Donno, D., Mainetti, L., Palano, L., Patrono, L., Stefanizzi, M.L., & Tarricone, L. (2015). An IoT-Aware Architecture for Smart Healthcare Systems. ● Chen, M., Mao, S., & Liu, Y. (2014). Big data: A survey. ● Chen, M., Tian, Y., Fortino, G., Zhang, J., & Humar, I. (2018). Cognitive Internet of Vehicles. ● Fernández, A., del Río, S., López, V., Bawakid, A., del Jesus, M.J., Benítez, J.M., & Herrera, F. (Year not provided). Big Data with Cloud Computing: An insight on the

	<p>computing environment, MapReduce, and programming frameworks.</p> <ul style="list-style-type: none"> ● Chen, M., Qian, Y., Hao, Y., Li, Y., & Song, J. (2018). Data-Driven Computing and Caching in 5G Networks: Architecture and Delay Analysis. ● Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., & Hassabis, D. (2016). Mastering the game of Go with deep neural networks and tree search.
<p>3.14. Mabkhot, M.M., Al-Ahmari, A.M., Salah, B., & Alkhalefah, H. (2018). Requirements of the smart factory system: A survey and perspective.</p>	<ul style="list-style-type: none"> ● Schumacher, A., Erol, S., & Sih, W. (2016). A Maturity Model for Assessing Industry 4.0 Readiness and Maturity of Manufacturing Enterprises. ● Liu, X.F., Shahriar, M.R., Al Sunny, S.M.N., Leu, M.C., & Hu, L. (2017). Cyber-physical manufacturing cloud: Architecture, virtualization, communication, and testbed. ● Lu, Y. (2017). Industry 4.0: A survey on technologies, applications and open research issues. ● Gokalp, M.O., Kayabay, K., Akyol, M.A., Eren, P.E., & Kocyigit, A. (2016). Big data for Industry 4.0: A conceptual framework. ● Lee, J., Bagheri, B., & Kao, H.-A. (2015). A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems. ● Qin, J., Liu, Y., & Grosvenor, R. (2016). A Categorical Framework of Manufacturing for Industry 4.0 and beyond. ● Turner, C.J., Hutabarat, W., Oyekan, J., & Tiwari, A. (2016). Discrete Event Simulation and Virtual Reality Use in

	<p>Industry: New Opportunities and Future Trends.</p> <ul style="list-style-type: none"> ● Wang, S., Wan, J., Li, D., & Zhang, C. (2016). Implementing Smart Factory of Industrie 4.0: An Outlook. ● Wang, S., Wan, J., Zhang, D., Li, D., & Zhang, C. (2016). Towards smart factory for industry 4.0: A self-organized multi-agent system with big data based feedback and coordination. ● Lin, Y.-C., Hung, M.-H., Huang, H.-C., Chen, C.-C., Yang, H.-C., Hsieh, Y.-S., & Cheng, F.-T. (2017). Development of Advanced Manufacturing Cloud of Things (AMCoT)-A Smart Manufacturing Platform.
<p>3.18. Kumar, N.M., & Mallick, P.K. (2018). The Internet of Things: Insights into the building blocks, component interactions, and architecture layers.</p>	<ul style="list-style-type: none"> ● Bandyopadhyay, D., & Sen, J. (2011). Internet of things: Applications and challenges in technology and standardization. ● Ray, P.P. (2018). A survey on Internet of Things architectures. ● Atzori, L., Iera, A., & Morabito, G. (2010). The Internet of Things: A survey. ● Zanella, A., Bui, N., Castellani, A., Vangelista, L., & Zorzi, M. (2014). Internet of things for smart cities. ● Xu, L.D., He, W., & Li, S. (2014). Internet of things in industries: A survey. ● Sethi, P., & Sarangi, S.R. (2017). Internet of Things: Architectures, Protocols, and Applications. ● Khan, R., Khan, S.U., & Zaheer, R., Khan, S. (2012). Future internet: The internet of things architecture, possible applications and key challenges. ● Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of

	<p>Things (IoT): A vision, architectural elements, and future directions.</p> <ul style="list-style-type: none"> ● Bonomi, F., Milito, R., Natarajan, P., & Zhu, J. (2014). Fog computing: A platform for internet of things and analytics. ● Bonomi, F., Milito, R., Zhu, J., & Addepalli, S. (2012). Fog computing and its role in the internet of things.
<p>3.20. Eisele, S., Mardari, I., Dubey, A., & Karsai, G. (2017). RIAPS: Resilient information architecture platform for decentralized smart systems.</p>	<ul style="list-style-type: none"> ● Lopez, P.G., Montresor, A., Epema, D., Datta, A., Higashino, T., Iamnitchi, A., Barcellos, M., ... Riviere, E. (2015). Edge-centric computing: Vision and challenges. ● Stubbs, J., Moreira, W., Dooley, R. (2015). Distributed Systems of Microservices Using Docker and Serfnode. ● Cirani, S., Davoli, L., Ferrari, G., Leone, R., Medagliani, P., Picone, M., Veltri, L. (2014). A scalable and self-configuring architecture for service discovery in the Internet of Things.
<p>3.24. Jacob, P.M., Mani, P. (2018). Software architecture pattern selection model for Internet of Things based systems.</p>	<ul style="list-style-type: none"> ● Bahga, A., Madiseti, V. (2015). Internet of Things: A Hands on Approach. ● Shaw, M., Garlan, D. (1996). Software Architecture: Perspectives on An Emerging Discipline. ● Mall, R. (2014). Fundamentals of Software Engineering. ● Jansen, A., Bosch, J. (2005). Software architecture as a set of architectural design decisions. ● Pressman, R.S. (2005). Software Engineering –A Practitioner'S Approach. ● Sommerville, I. (2017). Software Engineering.

<p>3.25. Bednar, P.M., Welch, C. (2020). Socio-Technical Perspectives on Smart Working: Creating Meaningful and Sustainable Systems.</p>	<ul style="list-style-type: none"> ● Bateson, G. (1972). Steps to an ecology of mind. ● Baumeister, R.F., Leary, M.R. (1995). The Need to Belong: Desire for Interpersonal Attachments as a Fundamental Human Motivation. ● Argyris, C., Schön, D.A. (1974). Theory in Practice: Increasing Professional Effectiveness. ● Berger, P.L., Luckmann, T. (1967). The social construction of reality: A treatise in the sociology of knowledge. ● Bertalanffy, L.V. (1969). General System Theory. ● Braverman, H. (1974). Labor and monopoly capital: The degradation of work in the twentieth century. ● Checkland, P.B. (1999). Systems thinking, systems practice: A 30-year retrospective. ● Chesbrough, H. (2003). Open innovation: The new imperative for creating and profiting from technology. ● Davenport, L., Prusak, L. (2000). Working knowledge: How Organizations manage what they know. ● Lee, J., Bagheri, B., Kao, H.-A. (2015). A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems.
<p>3.26. Muhammad, G., Alhamid, M.F., Long, X. (2019). Computing and processing on the edge: Smart pathology detection for connected healthcare.</p>	<ul style="list-style-type: none"> ● Chen, M., Li, W., Hao, Y., Qian, Y., Humar, I. (2018). Edge cognitive computing based smart healthcare system. ● Shi, W., Cao, J., Zhang, Q., Li, Y., Xu, L. (2016). Edge Computing: Vision and Challenges. ● ● Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D.,

	<p>Erhan, D., Rabinovich, A. (2015). Going deeper with convolutions.</p>
<p>3.16. Shafique, K., Khawaja, B.A., Sabir, F., Qazi, S., Mustaqim, M. (2020). Internet of things (IoT) for next-generation smart systems: A review of current challenges, future trends and prospects for emerging 5G-IoT Scenarios.</p>	<ul style="list-style-type: none"> ● Atzori, L., Iera, A., Morabito, G. (2010). The Internet of Things: A survey. ● Osseiran, A., Boccardi, F., Braun, V., Kusume, K., Marsch, P., Maternia, M., Queseth, O., Fallgren, M. (2014). Scenarios for 5G mobile and wireless communications: The vision of the METIS project. ● Yaqoob, I., Ahmed, E., Hashem, I.A.T., Ahmed, A.I.A., Gani, A., Imran, M., Guizani, M. (2017). Internet of Things Architecture: Recent Advances, Taxonomy, Requirements, and Open Challenges. ● Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. ● Kaur, N., Sood, S.K. (2017). An Energy-Efficient Architecture for the Internet of Things (IoT). ● Zhou, J., Leppanen, T., Harjula, E., Ylianttila, M., Ojala, T., Yu, C., Jin, H. (2013). CloudThings: A common architecture for integrating the Internet of Things with Cloud Computing. ● Cirani, S., Davoli, L., Ferrari, G., Leone, R., Medagliani, P., Picone, M., Veltri, L. (2014). A scalable and self-configuring architecture for service discovery in the Internet of Things. ● Lee, I., Lee, K. (2015). The Internet of Things (IoT): Applications, investments, and challenges for enterprises. ● Lloret, J., Tomas, J., Canovas, A., Parra, L. (2016). An Integrated IoT Architecture for Smart Metering.

	<ul style="list-style-type: none">• Zanella, A., Bui, N., Castellani, A., Vangelista, L., Zorzi, M. (2014). Internet of Things for Smart Cities.• Gupta, A., Jha, R.K. (2015). A Survey of 5G Network: Architecture and Emerging Technologies.
--	---

II PIELIKUMS: IOS mobilās pakotnes koda fragments

```
// ApplePayViewController.swift
import UIKit
import PassKit
import Stripe

// Protocol used for sending data back to product view
protocol ApplePayControllerDelegate: AnyObject {
    func applePayFinishedWithCheckout(paymentSuccess: Bool?)
}

class ApplePayViewController: UIViewController,
PKPaymentAuthorizationViewControllerDelegate, STPAuthenticationContext {
    func authenticationPresentingViewController() -> UIViewController {
        return self
    }

    var selectedVariant: PurpleNode?
    var checkout: CheckoutResponse?
    var paymentStatus: PaymentStatus?
    var paymentSuccess: Bool?
    var tag: String?
    weak var delegate: ApplePayControllerDelegate? = nil

    override func viewDidLoad() {
        super.viewDidLoad()
        // Do any additional setup after loading the view.
    }

    override func viewWillAppear(_ animated: Bool) {
        super.viewWillAppear(animated)

        self.purchase()
    }

    func paymentAuthorizationViewControllerDidFinish(_ controller:
PKPaymentAuthorizationViewController) {
        controller.dismiss(animated: true, completion: self.dismiss)
    }
}
```

```
func dismiss() {  
    self.dismiss(animated: true, completion: nil)  
    delegate?.applePayFinishedWithCheckout (paymentSuccess:  
paymentSuccess)  
}
```

III PIELIKUMS: Android mobilās pakotnes koda fragments

```
class MonetizrSdk {

    companion object {
        var debuggable: Boolean = false
        var dynamicApiKey: String? = null
        private var playerId: String? = null
        private var lockedProduct: Boolean = false
        private var initialLaunch: Boolean = true
        private var progressDialog: AlertDialog? = null

        /**
         * Show product with specified tag for specific user
         *
         * @param String productTag          Product tag that is
provided to customer
         * @param Boolean locked_product    Determines if this
product has to be displayed as locked
         * @param String player_id          Player ID
         */
        fun showProductForTag(productTag: String, locked_product: Boolean =
false, player_id: String? = null) {
            try {
                val context = ApplicationProvider.application as Context
                val activity = ActivityProvider.currentActivity

                if (activity == null) {
                    logError("Activity context is null")
                    return
                }

                if (isNetworkAvailable(context) == false) {
                    logError("Did not have internet access")
                    ErrorMessageBuilder.makeDialog(activity,
context.getString(R.string.no_network)).show()
                    return
                }

                var apiKey = ConfigHelper.getConfigValue(context,
Parameters.RAW_API_KEY)
```

```

        // Use specified api key if it is passed to method
        if (dynamicApiKey?.equals(null) == false) {
            apiKey = dynamicApiKey.toString()
        }

        lockedProduct = locked_product

        if (player_id?.equals(null) == false) {
            playerId = player_id.toString()
        }

        val endpoint = ConfigHelper.getConfigValue(context,
Parameters.RAW_API_ENDPOINT)

        progressDialog =
ProgressDialogBuilder.makeProgressDialog(activity)

        showProgressBar()
        requestProductInformation(activity, productTag, endpoint,
apiKey)
        sendTelemetryInfo(activity)

    } catch (e: Exception) {
        logError(e)
    }
}

private fun sendTelemetryInfo(context: Context) {
    Telemetry.sendDeviceInfo()

    if (initialLaunch) {
        Telemetry.session(ApplicationProvider.sessionStart)
        Telemetry.sendFirstImpression()
        initialLaunch = false
    }

    val monetizrSdkPreference =
PreferenceManager.getDefaultSharedPreferences(context)

```

```

        val isFirstRun =
monetizrSdkPreference.getBoolean(Parameters.IS_FIRST_RUN, true)

        if (isFirstRun) {
            val editor = monetizrSdkPreference.edit()
            editor.putBoolean(Parameters.IS_FIRST_RUN, false)
            editor.apply()

            Telemetrics.sendFistRun()
        }

        val lastUpdateVersion =
monetizrSdkPreference.getInt(Parameters.LAST_UPDATE_VERSION,
BuildConfig.VERSION_CODE)
        val currentVersion = BuildConfig.VERSION_CODE

        if (lastUpdateVersion == currentVersion) {
            val edit = monetizrSdkPreference.edit()
            edit.putInt(Parameters.LAST_UPDATE_VERSION,
lastUpdateVersion)
            edit.apply()
        } else if (lastUpdateVersion < currentVersion) {
            Telemetrics.update()
        }
    }
}

```

IV PIELIKUMS: Unity mobilās pakotnes koda fragments

```
namespace Monetizr
{
    public delegate void MonetizrErrorDelegate(string msg);

    public delegate void MonetizrOrderDelegate(Product p);

    public class MonetizrMonoBehaviour : MonoBehaviour
    {
        private MonetizrSettings _settings;

        /// <summary>
        /// Functions subscribed to this delegate are called whenever
something
        /// calls <see cref="ShowError(string)"/>.
        /// </summary>
        public MonetizrErrorDelegate MonetizrErrorOccurred;

        /// <summary>
        /// Functions subscribed to this delegate are called whenever a
successful order is done. Do note that
        /// at the moment this is supported by Big Screen, Android with
UGUI and iOS (only with Apple Pay).
        /// </summary>
        public MonetizrOrderDelegate MonetizrOrderConfirmed;

        private GameObject _currentPrefab;
        private MonetizrUI _ui;
        private string _baseUrl = "https://api3.themonetizr.com/api/";
        private string _language;
        private string _playerId;
        private string _overrideAccessToken = "";

        /// region Initialization and basic features

#if UNITY_IOS && !UNITY_EDITOR && MONETIZR_IOS_NATIVE
        [DllImport("__Internal")]
        extern static private void objCinitMonetizr(string token);

        [DllImport("__Internal")]
```

```

extern static private void objCshowProductForTag(string tag);

[DllImport("__Internal")]
extern static private void objCinitMonetizrPlayerId(string
playerId);

[DllImport("__Internal")]
extern static private void objCinitMonetizrApplePay(string
merchantId, string companyName);

[DllImport("__Internal")]
extern static private void objCsetMonetizrTestMode(bool on);

public void iOSPluginError(string message) {
    ShowError(message);
}

public void iOSPluginPurchaseDelegate(string tag) {
    if (MonetizrOrderConfirmed != null)
        GetProduct(tag, x => MonetizrOrderConfirmed(x));
}
#endif

internal void Init(MonetizrSettings settings)
{
    _settings = settings;

    // Privatethere is no need to switch access token mid-session.
    DontDestroyOnLoad(gameObject);

#if UNITY_IOS && !UNITY_EDITOR && MONETIZR_IOS_NATIVE
    objCinitMonetizr(AccessToken);
    if(_settings.applePay)
    {
        objCinitMonetizrApplePay(_settings.applePayMerchantId,
_settings.applePayCompanyName);
        if (_settings.applePayTestMode)
            objCsetMonetizrTestMode(true);
    }
#endif

    CreateUIPrefab();

```

```

        Telemetrics.ResetTelemetricsFlags();
        Telemetrics.RegisterSessionStart();
        Telemetrics.SendDeviceInfo();

        SetPlayerId("AABB01010101NotSet");
    }
    /// <summary>
    /// Set a unique identifier for this player. This ID will be used
for handling
    /// claim orders and other personalized offers.
    /// </summary>
    /// <param name="newId"></param>
    public void SetPlayerId(string newId)
    {
        _playerId = newId;
#if UNITY_IOS && !UNITY_EDITOR && MONETIZR_IOS_NATIVE
        objCinitMonetizrPlayerId(_playerId);
#endif
    }
}

```


V PIELIKUMS: API koda fragments

```
class OfferViewSet(viewsets.ModelViewSet):
    """
    list: Return a list of offers for specified application
    """
    queryset = Offer.objects.all()
    serializer_class = OfferListSerializer
    permission_classes = (permissions.IsAuthenticated, IsStoreOwner,
TokenHasScope)
    http_method_names = ['get']
    required_scopes = ['read']
    filterset_class = OfferFilter

    def get_queryset(self):
        user = self.request.user
        application = self.request.auth.application

        return self.queryset.filter(store__user=user,
store__applications__pk__in=[application.id]).order_by('id')

    def get_serializer_class(self):
        serializer_class = self.serializer_class

        return serializer_class

# Add mixpanel event logging
def list(self, request, *args, **kwargs):
    store_api_request(request)

    user = self.request.user
    team_owner = user.team_owner
    application = self.request.auth.application

    # Temporary local storage to save the last shown segment
    local_cache = caches['local']

    # Random segments enabled
    random_segments = user.team_owner.random_segments
```

```

        # All of the segments created for this application and this user,
only with offers
        segments = Segment.objects.filter(testmode=application.test_app,
created_by=team_owner,
deleted_at__isnull=True).exclude(segment_offers__isnull=True).order_by('id'
)

        # Split segments in two list - with and without geolocation
segments_with_geolocation = list(filter(lambda segment:
segment.country, list(segments)))
segments_without_location = list(filter(lambda segment: not
segment.country, list(segments)))

        # A list with segments that are shown to the player
segments_shown = list()

        # There are segments attached to this application, and segments
have offers, change who can see them
if segments:
    # If random segments are enabled
    # Exclude the offers from the segments that should not be shown
offers_to_exclude = []

    if random_segments:

        if len(segments_without_location) > 1:

offers_to_exclude.extend(self.exclude_segment_offers(segments_without_locat
ion, local_cache, 'last_shown_segment_without_location', request,
application, segments_shown))
        else:
            # Segment was shown
            segments_shown.append(segments_without_location[0])

        if len(segments_with_geolocation) > 0:
            # User location specific segments
            available_segments_for_the_player = []

            if (len(segments_with_geolocation) == 1):
                # If there is only one segment with geolocation,
then just exclude the offers if those are unavailable in specific region

```

```

offers_to_exclude.extend(self.exclude_location_unavailable_offers(request,
segments_with_geolocation, application, segments_shown))
    else:
        # If there are multiple segments, then it is
required to remove those segments that are unavailable for specific player
        # When offers from unavailable regions are removed,
then
        for segment in segments_with_geolocation:
            if
(targeting.segmentUnavailableInRegion(segment, request, application)):
                offers =
list(segment.segment_offers.values_list('offer_id', flat=True))
                for offer in offers:
                    offers_to_exclude.append(offer)
            else:
                # Create a list with segments that are
relevant for random geolocation segment selection
available_segments_for_the_player.append(segment)

        # Check if there are segments that have to be
randomly switched between

        if (len(available_segments_for_the_player) > 0):
            # Switch between the segments, do not respect
the country, state, city, if they are available for this player, then
switch between them

offers_to_exclude.extend(self.exclude_segment_offers(available_segments_for
_the_player, local_cache, 'last_shown_random_segment_with_location',
request, application, segments_shown))

    else:
        # Respect geolocation specific segments and exclude offers
that are unavailable for specific player
        if (len(segments_with_geolocation) > 0):

offers_to_exclude.extend(self.exclude_location_unavailable_offers(request,
segments_with_geolocation, application, segments_shown))

```

```

        # Add the segments that were show to the player
        for segment in segments_without_location:
            segments_shown.append(segment)

    # Default queryset, with excluded offers from the segments that
    should be randomly showed for players
    queryset = self.get_queryset()
    queryset = queryset.exclude(pk__in=offers_to_exclude)

    # Segments that were shown, add then to the mixpanel data
    mixpanel_segments_titles = []
    mixpanel_segments_ids = []
    for segment in segments_shown:
        mixpanel_segments_titles.append(segment.title)
        mixpanel_segments_ids.append(segment.id)

    # Keep the pagination working
    page = self.paginate_queryset(queryset)
    if page is not None:
        serializer = self.get_serializer(page, many=True)
        store_offer_list_request(self.request, {'segments':
mixpanel_segments_titles, 'segments_ids': mixpanel_segments_ids })
        return self.get_paginated_response(serializer.data)

    serializer = self.get_serializer(queryset, many=True)
    store_offer_list_request(self.request, {'segments':
mixpanel_segments_titles, 'segments_ids': mixpanel_segments_ids })
    return Response(serializer.data)

    # By default, if none of cases are relevant, return all segments
    all offers
    store_offer_list_request(self.request)
    return super().list(request, *args, **kwargs)

```

VI PIELIKUMS: Mašīnmācīšanās algoritma fragments

```
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, f1_score, precision_score,
recall_score, roc_auc_score, confusion_matrix
import numpy as np
import pandas as pd
from imblearn.over_sampling import SMOTENC, RandomOverSampler
from sklearn.dummy import DummyClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import AdaBoostClassifier, RandomForestClassifier
import seaborn as sns
import matplotlib.pyplot as plt

from preprocess import categories

output_dir = 'data'
data_template = 'all_{}.csv'
metrics_file = 'tested_models.txt'

def create_confusion_matrix(labels_test, labels_pred, model_name,
balance_name):
    plt.figure()
    conf_mat = confusion_matrix(labels_test, labels_pred)
    sns.set(font_scale=1)
    sns.heatmap(conf_mat / np.sum(conf_mat), annot=True, fmt='.2%',
cmap='Blues')
    plt.ylabel('True labels')
    plt.xlabel('Predicted labels')
    plt.savefig("plots/{}_{}.png".format(model_name, balance_name))

if __name__ == '__main__':
    data = pd.read_csv(os.path.join(output_dir,
data_template.format('data')))
    labels = pd.read_csv(os.path.join(output_dir,
data_template.format('labels')))

    print('Total purchases {}'.format(np.sum(labels)))

    data_train, data_test, labels_train, labels_test =
train_test_split(data, labels, test_size=0.30, random_state=42)
```

```

    categorical_features = [any([column.startswith(prefix) for prefix in
categories] for column in data_train.columns)]
    data_train = data_train.values
    data_test = data_test.values
    labels_train = labels_train.values
    labels_test = labels_test.values

    models = {
        'dummy_classifier': DummyClassifier(random_state=50),
        'logistic_regression': LogisticRegression(verbose=1,
random_state=42),
        'random_forest_classifier': RandomForestClassifier(verbose=1,
random_state=42),
        'adaboost_classifier': AdaBoostClassifier(random_state=42)
    }

    balancing_methods = {
        'SMOTE_NC': SMOTENC(categorical_features=categorical_features,
random_state=4242),
        'Random OverSampling': RandomOverSampler(random_state=42)
    }

    for balance_name, oversampler in balancing_methods.items():
        print('Before {} Labels in training: {}'.format(balance_name,
Counter(labels_train.ravel()))
        data_train, labels_train = oversampler.fit_resample(data_train,
labels_train)
        print('After {} Labels in training: {}'.format(balance_name,
Counter(labels_train.ravel()))

    for model_name, model in models.items():
        model.fit(data_train, labels_train)
        labels_pred = model.predict(data_test)
        predicted_probabilities = model.predict_proba(data_test)

        with open(metrics_file, 'a') as fd:
            fd.write('Model: {}\n'.format(model))
            fd.write('Balancing method: {}\n'.format(oversampler))
            fd.write('Accuracy:
{:4f}\n'.format(accuracy_score(labels_test, labels_pred)))

```

```
        fd.write('F1: {:.4f}\n'.format(f1_score(labels_test,
labels_pred)))
        fd.write('Precision:
{:.4f}\n'.format(precision_score(labels_test, labels_pred)))
        fd.write('Recall: {:.4f}\n'.format(recall_score(labels_test,
labels_pred)))
        fd.write('ROC AUC:
{:.4f}\n\n'.format(roc_auc_score(labels_test, labels_pred)))

        create_confusion_matrix(labels_test, labels_pred, model_name,
balance_name)
```

VII PIELIKUMS: Datu kopas noteikšanas fragments

P5.1. tabula. Datu kopas noteikšanas fragments (autora veidota)

Device Data	Device	<i>iPhone10,6</i>	Automatically
	OS version	<i>9,1</i>	Automatically
	Region	<i>en_US</i>	Automatically
	Publisher	<i>Grow Pixel</i>	Automatically
	Date of game install with Monetizr SDK		Automatically
	Total number of players	<i>122617</i>	Automatically*
	Date of game update with Monetizr SDK		Automatically*
	Time from game load to 1st Monetizr Impression		Automatically*
	Time from 1st Monetizr Impression to 1st Monetizr reward click		Automatically*
Encounters	Impressions shown to user / monetizr produkti		Automatically*
	Impressions above 3 seconds		Automatically*
	Dismissed Impressions (below 3 seconds)		Automatically*
	Dismissal clicks		Automatically*
	Clicks on Monetizr (broad) / ne uz produkta, bet uz loga		Automatically*
	Clicks on Monetizr (reward) /		Automatically*
Game Data	Category	<i>Games: Puzzle</i>	User input
	Sub-category	<i>Games: Strategy</i>	User input
Design analytics	Total Number of in-game Monetizr triggers		User input
	Type of Monetizr trigger		User input
	Game level name with Monetizr encounter enabled		User input
	Trigger tag: violence, puzzle solved, puzzle unsolved, dialogue, game won, game over...		User input

	Difficulty level name		User input
	Difficulty level estimation		User input
	Game completion status	55%	User input
	Progressive game trigger list		User input
User behaviour data	Game progress	25	User input
	Time from 1st Monetizr Impression to Monetizr Account creation		Backend
	Time from 1st Monetizr Impression to 1st purchase / laiks kamēr nopērk		Backend
	Time from 1st Monetizr click to 1st purchase / laiks		Backend
	Apple Pay satus	<i>Active</i>	Native code required
	Google Wallet status		Native code required
	Reward claims		Need more info
	Reward claim types		Need more info
Gamer profile (SDK v3)	Age	23	Unity not allowing to read
	Gender	<i>Male</i>	Unity not allowing to read
	Video views		Videos only
	Video completion (25%)		Videos only
	Video completion (50%)		Videos only
	Video completion (75%)		Videos only
	Video completion (100%)		Videos only
	Device app list		Unity not allowing to read
	Gamecenter - Number of games installed (with consent)	8	Won't implement
	Background music name prior to Monetizr encounter		Won't implement
	Background music download for Monetizr analytics enabled (TRUE/FALSE)		Won't implement
	Sound effect name on Monetizr encounter		Won't implement

	Sound effect download for Monetizr analytics enabled (TRUE/FALSE)		Won't implement
	High RAM usage indicator		Won't implement
User purchase behaviour	IAP spending amount	<i>8,97</i>	Won't implement
	IAP spending amount through Monetizr / in-app purchases		Won't implement
	Number of IAP	<i>3</i>	Won't implement
	Number of IAP through Monetizr / in-app purchases		Won't implement
	Date of IAP		Won't implement
	Date of IAP through Monetizr		Won't implement
	Purchase currency		Won't implement
	Purchase currency through Monetizr		Won't implement
	IAP Product		Won't implement
	IAP Product through Monetizr		Won't implement
	Color map sample		Not for Plugin
	Game funnel trigger list		Not for Plugin
	Game time	<i>455</i>	Not for Plugin
	Last session / spēlētāja	<i>1/1/2019</i>	Not for Plugin
	Session time	<i>12</i>	Not for Plugin
	Start of game session		Not for Plugin
	End of game session		Not for Plugin
	Number of dismissals		Not for Plugin
	Number of non-dismissals		Not for Plugin
	Dismissal ratio		Not for Plugin
Customer info (only after purchase)	City	<i>Chicago</i>	Not for Plugin
	State	<i>IL</i>	Not for Plugin
	Country	<i>USA</i>	Not for Plugin
	Spending amount	<i>13,99</i>	Not for Plugin

	Size	<i>M</i>	Not for Plugin
	Product ordered	<i>T-Shirt</i>	Not for Plugin
	Number of games played with Monetizr encounters		Not for Plugin
Tracking adjustment	Number of disabled tracking options		Not for Plugin
	Name of disabled tracking option		Not for Plugin
	Date of tracking adjustment		Not for Plugin
All user behaviour data	see list above		Not for Plugin
All encounter data	see list above		Not for Plugin
All purchase behaviour data	see list above		Not for Plugin
	Dates when Game featured on the app store	<i>23/11/2018, 22/11/2018</i>	Not for Plugin
	Sentiment of Reviews	<i>5</i>	Not for Plugin
Game Data	# of Ratings on the app store	<i>6,04k</i>	Not for Plugin
	Average Rating on the app store	<i>4,8</i>	Not for Plugin
	Languages	<i>English, Czech, Danish, Dutch, Finnish, Spanish</i>	Not for Plugin
	Size	<i>165.2 MB</i>	Not for Plugin
	First release date	<i>20/11/2018</i>	Not for Plugin
	Latest update	<i>23/11/2018</i>	Not for Plugin
	Version number	<i>13.45.7</i>	Not for Plugin
	Price	<i>\$0.99</i>	Not for Plugin
	App store countries	<i>USA, Germany</i>	Not for Plugin
	Overall rank	<i>12345</i>	Not for Plugin
	Category Rank	<i>13</i>	Not for Plugin
	Subcategory Rank	<i>45</i>	Not for Plugin
	Top Grossing Rank (estimate)	<i>433</i>	Not for Plugin
	Estimated Total Revenue	<i>\$1.005.000</i>	Not for Plugin
	Estimated In App Purchase Revenue	<i>\$805.00</i>	Not for Plugin

	Estimated Advertising Revenue	<i>\$200.00</i>	Not for Plugin
	Estimated Paid Download Revenue	<i>\$0</i>	Not for Plugin
	Estimated Average Revenue Per User	<i>\$2.36</i>	Not for Plugin
	Estimated daily downloads	<i>20518</i>	Not for Plugin
	Estimated Daily Active Users	<i>300000</i>	Not for Plugin
	Estimated Monthly Active Users	<i>600000</i>	Not for Plugin
	Engagement % (estimate)	<i>25%</i>	Not for Plugin
	30 Day Retention (estimate)	<i>10%</i>	Not for Plugin
	# of Sessions Per User (estimate)	<i>5</i>	Not for Plugin
	Total # of Sessions (estimate)	<i>100000000</i>	Not for Plugin
	Average Session Length Per User (estimate)	<i>2,54</i>	Not for Plugin
	Total Time Spent In App (estimate)	<i>100000000</i>	Not for Plugin
	Last 24hr Rank Change	<i>45</i>	Not for Plugin
	Breakout potential (%)	<i>80.50%</i>	Not for Plugin
	% Male	<i>55</i>	Not for Plugin
	% Female	<i>45</i>	Not for Plugin
	Most Likely Age Of Users	<i>23</i>	Not for Plugin
	% Of Users 10-20 yrs. Old	<i>15</i>	Not for Plugin
	% Of Users 21-30 yrs. Old	<i>60</i>	Not for Plugin
	% Of Users 31-40 yrs. Old	<i>20</i>	Not for Plugin
	% Of Users 41-50 yrs. Old	<i>4</i>	Not for Plugin
	% Of Users 51+ yrs. old	<i>1</i>	Not for Plugin

VIII PIELIKUMS: Datu kopas paraugs

Pilna datu versija pēc pieprasījuma pieejama.

https://storage.cloud.google.com/monetizr-sql-dumps/device_test_data.xlsx

Time	CPU max	CPU average	RAM in use	Total RAM	RAM usage
16:06	2310	1456	968	3574	27
16:07	2310	1326	958	3574	27
16:08	2310	1085	1004	3574	28
16:09	2310	1131	1061	3574	30
16:10	2310	702	1084	3574	30
16:11	2310	1014	1076	3574	30
16:12	2310	1112	969	3574	27
16:13	2310	744	1007	3574	28
16:14	2310	924	999	3574	28
16:15	2310	1120	849	3574	24
16:16	2310	1568	825	3574	23
16:17	2310	403	816	3574	23
16:18	2310	1068	803	3574	22
16:19	2310	746	791	3574	22
16:20	2310	687	801	3574	22
16:21	2310	750	815	3574	23
16:22	2310	753	859	3574	24
16:23	2310	753	835	3574	23
16:24	2310	769	820	3574	23
16:25	2310	1096	819	3574	23
16:26	2310	403	839	3574	23
16:27	2310	1113	794	3574	22
16:28	2310	1112	815	3574	23
16:29	2310	922	838	3574	23
16:30	2310	887	820	3574	23
16:31	2310	687	791	3574	22
16:32	2310	750	835	3574	23
16:33	2310	1114	820	3574	23
16:34	2310	746	819	3574	23
16:35	2310	687	839	3574	23
17:34	2310	1610	878	3574	25
17:35	2310	1742	872	3574	24
17:36	2310	1417	920	3574	26
17:37	2310	1742	823	3574	23
17:38	2310	1551	820	3574	23
17:39	2310	806	826	3574	23
17:40	2310	958	821	3574	23
17:41	2310	1604	791	3574	22
17:42	2310	1059	882	3574	25
17:43	2310	1050	913	3574	26
17:44	2310	640	902	3574	25
17:45	2310	910	922	3574	26

P11.1.attēls. Datu kopas paraugs (autora veidots).

IX PIELIKUMS: Datu priekšapstrādes paraugs

```
def simplify_gpu(df):
    df['GPU frequency'] = df['GPU frequency'].fillna(-0.5) #fill missing
    bins = (-1, -0.4, 586, 1500)
    group_names = ['unknown', 'off', 'on']
    categories = pd.cut(df['GPU frequency'], bins, labels=group_names)
    df['GPU frequency'] = categories
    return df

def simplify_cpu(df):
    df['CPU average'] = df['CPU average'].fillna(-0.5) #fill missing
    bins = (-1, 0, 989, 1271, 1678, 3000)
    group_names = ['unknown', '1_quartile', '2_quartile', '3_quartile',
'4_quartile']
    categories = pd.cut(df['CPU average'], bins, labels=group_names)
    df['CPU average'] = categories
    return df

def simplify_ram(df):
    df['RAM in use'] = df['RAM in use'].fillna(-0.5) #fill missing
    bins = (-1, 0, 846, 936, 1008, 4096)
    group_names = ['unknown', '1_quartile', '2_quartile', '3_quartile',
'4_quartile']
    categories = pd.cut(df['RAM in use'], bins, labels=group_names)
    df['RAM in use'] = categories
    return df

def simplify_battery(df):
    df['Battery temperature'] = df['Battery temperature'].fillna(-0.5)
#fill missing
    bins = (-1, 0, 27, 28, 30, 40)
    group_names = ['unknown', '1_quartile', '2_quartile', '3_quartile',
'4_quartile']
    categories = pd.cut(df['Battery temperature'], bins,
labels=group_names)
    df['Battery temperature'] = categories
    return df
```

```

def simplify_bat_con(df):
df['Device_power_estimate_percentage']=
df['Device_power_estimate_percentage'].fillna(-0.5) #fill missing
    bins = (-1, 0, 1, 6, 10, 20)
    group_names = ['unknown', 'flight', 'low', 'medium', 'extensive']
    categories = pd.cut(df['Device_power_estimate_percentage'], bins,
labels=group_names)
    df['Device_power_estimate_percentage'] = categories
    return df

def simplify_janky_percent(df):
    df['Janky_percent'] = df['Janky_percent'].fillna(-0.5) #fill missing
    bins = (-1, 0, 0.14, 1)
    group_names = ['unknown', 'normal', 'on']
    categories = pd.cut(df['Janky_percent'], bins, labels=group_names)
    df['Janky_percent'] = categories
    return df

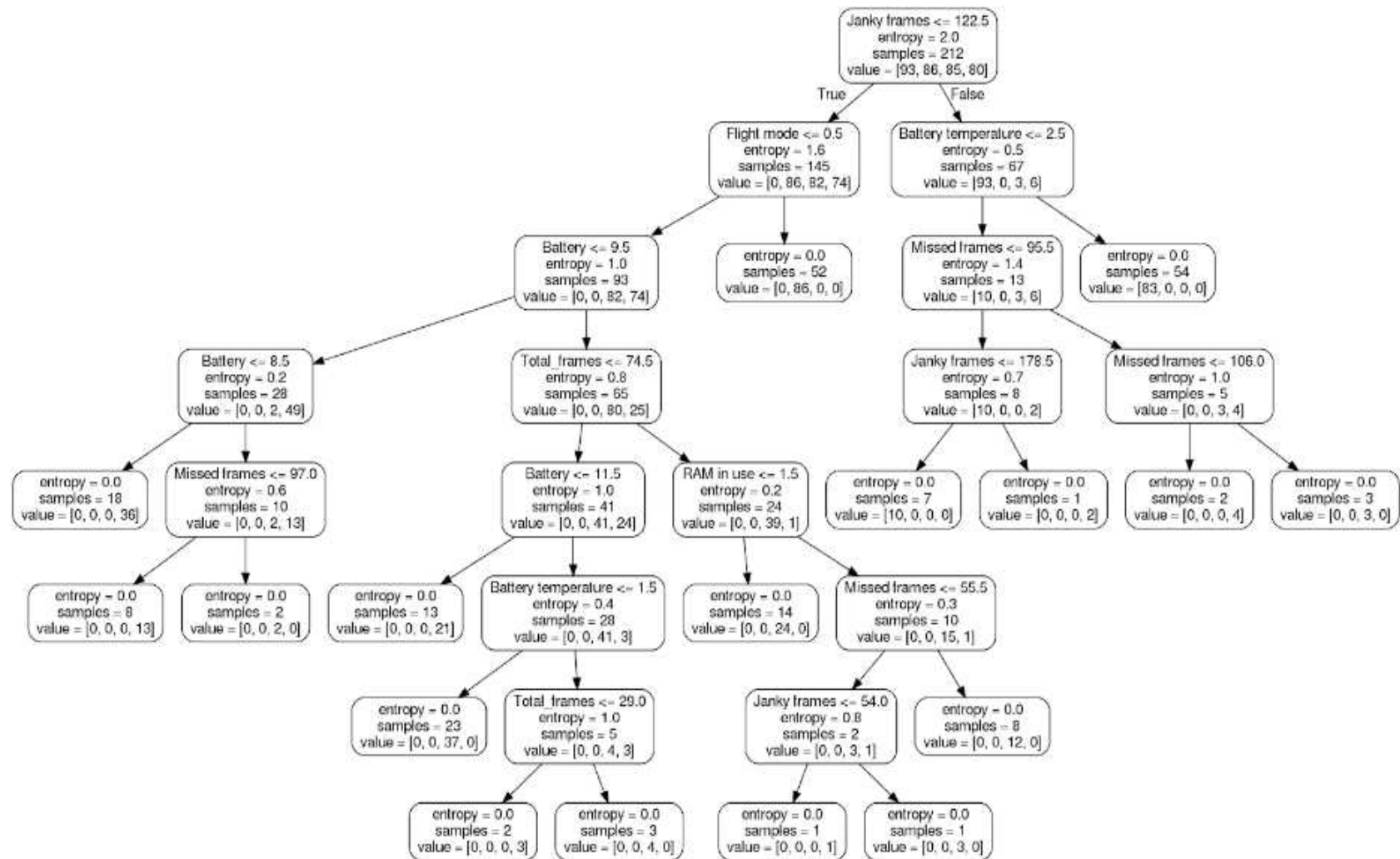
def drop_features(df):
    return df.drop(['OpenGL ES', 'Java VM', 'API level', 'Android version',
'CPU type', 'Test case', 'Time', 'CPU max', 'Total RAM', 'Battery status',
'GPU', 'GPU core speed', 'Device model', 'Wi-Fi On', 'Calculator on',
'Phone catalog on', 'Browser on', 'Facebook on', 'Instagram on', 'Mobile
banking on', 'Google Maps on','Youtube on', 'Asphalt nitro game on', 'RAM
usage', 'GPU load'], axis=1)

def transform_features(df):
    df = simplify_gpu(df)
    df = simplify_cpu(df)
    df = simplify_ram(df)
    df = simplify_battery(df)
    df = simplify_bat_con(df)
    df = simplify_janky_percent(df)
    df = drop_features(df)
    return df

data_train = transform_features(data_train)
data_test = transform_features(data_test)
data_test.head()
data_test.sample(5)

```

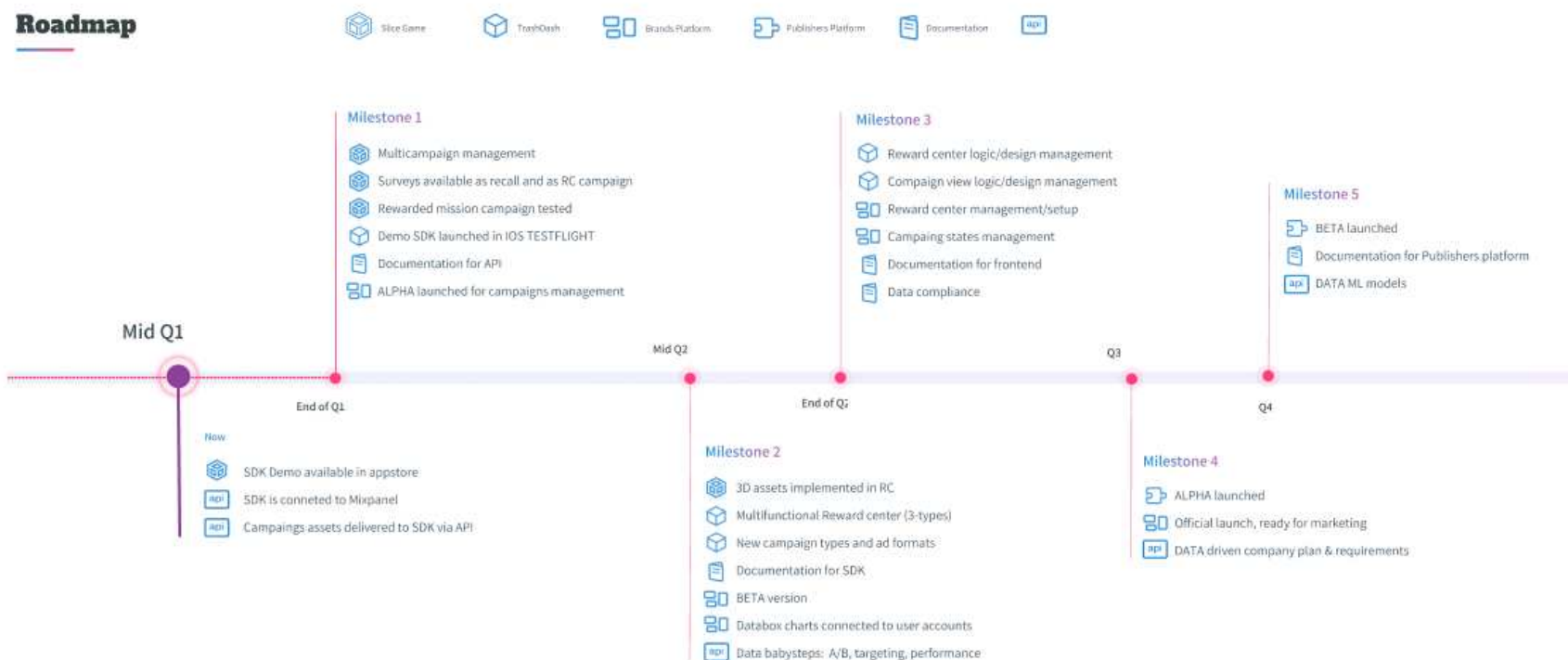
X PIELIKUMS: Lēmumu koka vizualizācija



P10.1.attēls. Lēmumu koka vizualizācija (autora veidots).

XI PIELIKUMS: Tehnoloģiju ceļveža paraugs

Product roadmap



P11.1. attēls. Tehnoloģiju ceļveža paraugs (autora veidots).

XII PIELIKUMS: Mašīnmācīšanās ietekme

Mašīnmācīšanās algoritmu modeļus parasti rada ņemot vērā aplūkojamās sfēras nozīmīgākos faktoros un izmantojot zināmos vēsturiskos datus. Pielietojot mašīnmācīšanās pieeju iespējams veidot algoritmus, kuri spēj izgūt zināšanas no liela apjoma lietotāju reakcijas datiem. Komunikāciju tehnoloģijām attīstoties, tās rada izaugsmi piedāvāto servisu klāstā un tiek radīti jauni lietošanas gadījumi, piemēram, viedās pilsētas, veselības monitorings, vieglie auto ar autopilotiem. Lielākā daļa no šiem lietojuma gadījumiem iekļauj pilnībā automatizētu komunikāciju starp iekārtām bez cilvēku starpniecības, sauktu par mašīna-mašīna komunikāciju (M2M) (Mehmood et al., 2015). M2M komunikācija atšķiras ar to, ka tai ir augstāks blīvums (angliski: traffic density), pakešu izmērs un servisa kvalitātes prasības. M2M noslogo mobilo tīklu pretēji kā to šobrīd izmanto viedtālruni, šāda veida komunikācija rada slodzi nevis lejupejošā (angliski: downlink), bet augšupejošā (angliski: uplink) plūsmā, kas salīdzinoši vienkārši var pārslogot esošos mobilos tīklus. Attīstoties šāda veida komunikācijas paveidam, nepieciešams attīstīt arī komunikāciju protokolus, kur zināmākie ir IEEE 802.15.4, 6LoWPAN, EIB/KNX, LON un BACnet (Mehmood et al., 2015). Šo protokolu izstrādes nolūks ir atslogot mobilos tīklus, izmantojot iekārtu fizisko tuvumu, piemēram, pieļaujot iekārtu savstarpēju komunikāciju un neiesaistot tīkla pamatelementus (vārtejas un komutatorus).

Mašīnmācīšanās pakāpeniski ir ieņēmusi būtisku lomu dažādās sociālās sfērās, piemēram, valodas apstrāde, dabīgās valodas sapratne, neirozinātne, lietu internets u.c (Zhou et al., 2017). Tradicionālie mašīnmācīšanās algoritmi šobrīd saskaras ar jauna veida izaicinājumiem, tādiem kā mērogojamība. Mašīnmācīšanās primāri atbild uz jautājumu kā izstrādāt datorsistēmu, kura spētu attīstīties balstoties uz pieredzi (Jordan & Mitchell, 2015).

Aplūkojot ražošanas nozari, uzņēmumi saskaras ar tādiem izaicinājumiem kā nepārtraukti pieaugoša ražošanas procesu sarežģītība (Villanueva et al., 2018). Datu apjoms, kurš rodas ražošanas procesā, pateicoties digitalizācijai ir pieaudzis gan apjomā, gan sarežģītībā. Datu analīze ne tikai satur pieejas, lai noteiktu iezīmes, kuras apraksta pagātni, bet arī mēģina veikt prognozes, lai noteiktu preventīvās darbības un uzlabotu ražošanas procesu (Villanueva et al., 2018).

Viens no mašīnmācīšanās uzdevumiem, kas iekļauj noteikta mainīgā prognozēšanu iepriekš neredzētos datos, tiek saukts par klasifikāciju. Daži no zināmākajiem klasifikācijas

paraugiem ir aizdevumu apstiprināšana, medicīniskās diagnozes, e-pastu filtrēšana un sporta spēļu rezultātu prognozes (Bunker & Thabtah, 2019).

Kā piemēru lietošanas gadījumam, kurā mašīnmācīšanās var radīt ieguvumus sabiedrībai kopumā, var minēt mašīnmācīšanās pielietojumu pilsētu jumtu analizēšanā, lai noteiktu to piemērotību saules paneļu uzstādīšanai (Mohajeri et al., 2018). Eksistē vairāki pētījumi, kuros veikta saules bateriju potenciāla noteikšana balstoties uz jumta formu. Tā, piemēram, vienā no pētījumiem kopā ar paraugu ņemšanas tehnoloģiju, tiek izmantota arī ģeogrāfiskās informāciju sistēmas bāzētu funkciju analīze, lai noteiktu jumtu saules paneļu uzstādes potenciālu (Wiginton et al., 2010). Citās pieejās tiek izmantoti aerofoto attēli un zemes virsmas lāzerskenējuma (LiDAR) dati, lai noteiktu jumtu ģeometrijas un izmērītu to potenciālu (Gooding et al., 2015). Arī šajos gadījumos mašīnmācīšanās algoritmu modeļi sniedz iespēju analizēt fotoattēlus informācijas apstrādei un potenciāla prognozei.

Liela daļa no mašīnmācīšanās algoritmu sistēmu ieviešanai patērētā laika parasti tiek izmantots datu priekšapstrādes procesu izveidē, kas rezultējas reprezentējamos un izmantojamajos datos (Baldi et al., 2017). Datu priekšapstrādes uzdevums ir risināt tādas datu problēmas kā pārmērība (angliski: redundancy), troksnis, heterogēnums, transformācijas, nesabalansētība un funkciju izvēle (Zhou et al., 2017).

Optimizācija ir jebkura MM algoritma pamatelements, optimizāciju iedala kombinatoriskajā un nepārtrauktajā (Sacha et al., 2017). Kombinatoriskā optimizācija ir resursietilpīga, ja datu kopa ir liela, jo datiem vēlams iegūt optimālu parametru skaitu, izejot cauri tikai vienu reizi (Xing et al., 2015).

Eksistējošie MM algoritmi var izmantot lielo datu pieejas, lai sasniegtu mērogojamību, piemēram, izveidot starpslāni, kurš spēj atkārtot esošo mācīšanās fāzi tā, lai to varētu izmantot lielo datu platformās kā Hadoop un Spark (Zhou et al., 2017). Apache Spark un Mahout ir divi atvērtā koda projekti, kuri atbalsta daudzus mērogojamus mašīnmācīšanās algoritmus (*Apache Spark*, 2021); (*Apache Mahout*, 2022). Liela daļa biežāk sastopamo MM algoritmu, iekļaujot klasifikācijas, regresijas un klasterēšanas, ir iekļautas Apache Spark un Mahout projektos.

Tādu problēmu risināšanā kā algoritmu sarežģītība un atmiņas ierobežotība tiek piedāvāts izmantot dalītu MM (Kozik et al., 2018). Dalītā MM, lai risinātu apstrādes ātruma problēmu, tiek vienlaikus izpildīta vairākās iekārtās vienlaikus, palielinot mācīšanās ātrumu un risinot datu zaudēšanas problēmu, pateicoties datu dublēšanai. GraphLab ir dalītais MM ietvars, kurš ļauj MM ekspertiem izstrādāt un izveidot efektīvus un mērogojamus dalītos MM algoritmus (Zhou et al., 2017).

Dziļā neironu tīklu bāzētā mācīšanās pieeja pakāpeniski kļūst par vienu no populārākajām. Neironu tīklu modeļi ir modeļu saime, kuru ierosinājuši bioloģiskie neironu tīkli, kurus veido savstarpēji savienoti neironi (Kozik et al., 2018). Dziļie neironu tīkli tiek uzverti kā neironu tīkli ar daudziem slēptiem slāņiem, kur katrs slānis piemēro nelineāru funkciju tās ievadam, lai izmainītu izvadu (Kozik et al., 2018). Dziļās neironu mācīšanās risinājumi ir tādi kā Theano, Caffe, Torch, Tensorflow, kuri izmanto iekārtas videokartes resursus, lai izpildītu mācīšanās procesu (Zhou et al., 2017). Liela apjoma dati ir viens no ierosinātajiem dziļajai mācīšanās pieejai. Dziļā mācīšanās parasti spēj atpazīt līdz pat 100 dažādām kategorijām.

MM priekš lieliem datiem rada izaicinājumus arī cilvēkiem, jo gandrīz neeksistē eksperti, kuriem būtu visas nepieciešamās zināšanas par MM, datu apstrādes pieejas tehnoloģijām, statistisko analīzi, pētāmo nozari, datubāzu tehnoloģijām, serveru konfigurāciju, programmēšanas valodām un saskarņu izstrādi. Tādēļ pakāpeniski notiek mēģinājumi izstrādāt informācijas sistēmas, kuras ir vieglāk saprast un lietot nespeciālistiem (Zhou et al., 2017). Lai MM kļūtu pieejamāka, tai nepieciešams kļūt vienkāršāk lietojamai un rezultātam tādām, lai gala lietotāji tos varētu interpretēt un novērtēt. Eksistējošo MM algoritmu sarežģītība var būt nomācoša nepieredzējušam lietotājam, jo šie lietotāji nevar saskatīt ieguvumus un izaicinājumus, kurus spēj sagādāt MM algoritma parametru, apmācības metodes un hiperparametru izvēles nozīme un ietekme uz algoritma izpildes laiku (Xing et al., 2015).

M2M komunikācijas procesā iegūtie dati var radīt arī privātuma problēmas, jo dati var saturēt personas informāciju, kuru tā nav atļāvusi izmantot, bet kura tur ir nokļuvusi, piemēram, fotoattēlu uzņemšanas procesā ielās, lai analizētu satiksmes intensitāti (Frias & Pérez Martínez, 2018). Šo datu drošības problēmu nepieciešamas risināt, lai personu dati nenokļūtu MM algoritmu procesā.

Viens no nopietnākajiem izaicinājumiem MM ir mācīšanās posma iterācijas izpildes ātrums, jo esošie ietvari nav pietiekami optimizēti MM algoritmu izpildei. Kā problēmu var minēt arī klasifikatoru skaitu - optimālu to skaita ieguvi izpildes laikā (Sacha et al., 2017). MM sarežģījumus rada arī mainīgas datu kopas, esošie MM algoritmi nav piemēroti darbam ar pieaugošu datu apjomu (Zhou et al., 2017). Mašīnmācīšanas algoritmi kā ieejas datus var apstrādāt tikai viena veida noteiktus datus, piemēram, ciparisku vai tekstuālu informāciju un attēlus, savukārt tehnoloģijām attīstoties, lietotāju reakcijas lietošanas gadījumu dati var būt dažāda veida un formātu (Zhai et al., 2014).

XIII PIELIKUMS: Tehnoloģiju ceļvedis

Tehnoloģiju ceļvedis ir paredzēts kā elastīgs plānošanas paņēmieni, lai atbalstītu stratēģisko un ilgtermiņa plānošanu, saskaņojot īstermiņa un ilgtermiņa mērķus ar konkrētiem tehnoloģiju risinājumiem. Tas ir plāns, kas attiecas uz jaunu produktu vai procesu izstrādi un var ietvert tehnoloģiju prognozēšanu vai tehnoloģiju izpēti, lai identificētu piemērotas jaunās tehnoloģijas (Phaal et al., 2004).

Balstoties uz pētnieku ieteikumiem (Lombardo et al., 2018) produkta, ietvara un tehnoloģiju ceļvedim jānodrošina sekojošais:

- organizācijas plāniem jābūt definētiem stratēģiskā kontekstā. Bieži produktu ceļvedis tiek vērsts uz rezultātiem, izlaižot kritisko kontekstu un mērķus. Rezultātā izstrādes komandas patērē daudz laika analizējot tirgus datus, klientu ieguldījumu, nosakot prioritātes, novērtējot, projektējot, veidojot arhitektūru, dažkārt neizskaidrojot pieņemtus lēmumus visām iesaistītajām pusēm. Tehnoloģiju ceļvedim jābūt iekļautam starp organizācijas misiju un detalizētiem izstrādes, izlaišanas un ieviešanas plāniem;
- tehnoloģijām jākoncentrējas uz pievienotās vērtības radīšanu organizācijai un tās klientiem. Bieži tehnoloģiju un produktu ceļvedis ir vairāk kā projekta plāns, kas vērsts uz efektīvu resursu izmantošanu, maksimālu efektivitāti un termiņu sasniegšanu. Daudzi detalizēti ceļveži izslēdz pievienotās vērtības definīciju un to, kas sagaidāms izstrādes rezultātā. Kritērijs, kuru bieži izmanto organizācijas vadība, lai novērtētu produktu izstrādes panākumus, ir tas vai komanda ir ievērojusi termiņus un darbojas saskaņā ar grafiku, nevis vai produkts nodrošina pievienoto vērtību;
- jaunu zināšanu iegūšanai un nepārtrauktam mācīšanās procesam jābūt daļai no tehnoloģiju izstrādes procesa. Izstrādes ceļveži bieži cenšas paredzēt noteiktu nākotnes stāvokli. Nozīmīgāk ir definēt pievienoto vērtību, klientu mērķus, klientu pieprasīto funkcionalitāti, prioritātes un klientu problēmas. Noteiktās pievienotās vērtības formulējums, kuru plānots nodrošināt, ir būtiskāks par informāciju kā tā tiks nodrošināta. Tehnoloģiju un produktu izstrādes laikā prasības un vēlmes var būtiski mainīties, tādēļ izstrādei un ieviešanai jākoncentrējas uz nepārtrauktu pielāgošanos un vērtības piegādi, nevis termiņiem;
- organizācijai jākoncentrējas uz vienotu prioritāšu kopu. Mārketinga un pārdošanas departamenti bieži pārdod kaut ko tādu, ko produkta izstrādes komandas neizstrādā.

Organizācijas, kuras nesaskaņo mārketinga, pārdošanas un produktu komandas vienotas prioritātes var zaudēt tirgus iespējas, var paiet mēneši, lai mārketinga un pārdošanas komandas saprastu, ko produktu komandas izstrādā un piegādā;

- tehnoloģiju ceļvedim nepieciešams klientus padarīt ieinteresētus par produkta attīstību un virzienu. Termiņu izpilde un plānoto funkciju piegāde negarantē ne tirgus atsaucību, ne organizācijas biznesa mērķu sasniegšanu. Atbilstošs eksperimentu apjoms var palīdzēt noteikt, ko nepieciešams izstrādāt, un metrikas, kuras tiks pielietotas novērtēšanai.

XIV PIELIKUMS: Monetization Solutions Inc. atsauksme



Reference about the research and development projects

Program: European Smart Growth Operational Program 2014-2020 with the participation of capital funds subgroup 1.3.1: Support for research and development projects for the development of conceptual projects - Bridge Alfa.

Research and development projects:


1. Using machine learning for improving monetization in mobile games;
2. Use of machine learning methods to predict the number of available computing resources in mobile devices.

The aim of the research projects was to develop models for retrieving knowledge from user feedback data that the company could use to make data-based decisions and improve the range of services offered by the company. The results of the projects were evaluated by independent auditors, who gave positive opinions. Mārtiņš Janševskis is a co-author of these projects and a research project manager.

During the development of the research project "Using Machine Learning to Improve Monetization in Mobile Games", a collaboration with the Polish company DLabs Sp. zo .o. was established. Mārtiņš worked in collaboration with DLabs CTO, Math Ph.D. and machine learning experts. The achieved results were a machine learning model to predict whether a specific device will be monetized.

In the process of the project "Using Machine Learning Methods to Predict the Number of Available Computing Resources in Mobile Devices", Mārtiņš Janševskis carried out an independent and individual research project in accordance with the standards set for the project and provided appropriate reports that ensured the successful implementation of the project.

Monetizing Solutions II, Inc.



(Martins Bratuskins, Founder, COO)

Date: ____21.06.2022____



Monetizr | 1995 Market St, Unit 808, San Francisco, CA

94103

Martins Bratuskins | COO & Co-Founder | martins@themonetizr.com



XV PIELIKUMS: P. Kolhe atsauksme

For Martins Jansevskis (MJ), Monetizr CTO, PhD cand.
Latvia

REFERENCE ON KNOWLEDGE DISCOVERY FROM FEEDBACK DATA FRAMEWORK

Santa Monica, California, United States

I was introduced to Martins by my friend from Techstars 17 as they were looking for a software engineer. During the following years we have discussed in depth the architectural structure of the knowledge discovery from user feedback data proposals. The framework can be used to create a scalable tech framework for a startup and it would be elastic to pivoting business changes.

Pushkar Kolhe

P.Kolhe

Software Engineer at Snapchat Inc.
Former Co-Founder & CTO of MoQuality
Georgia Institute of Technology, Computer Science