

Raport Științific și Tehnic

Identificare:

Contract de finanțare nr. 124/01/04.2020

Contractor: Universitatea de Vest din Timișoara (UVT)

Cod: ERANET-CHISTERA3-DIPET

Etapă 1: 1 Dec 2020 - 30 Noi 2021

Nume etapă: Prototip software

Autori ai raportului și membrii ai proiectului:

Lect. Dr. Gabriel Iuhasz (Co-PI)

Dr. Silviu Panica

Lect. Dr. Marian Neagul

Asist.cercetare Alexandru Munteanu

Director proiect:

Prof. Dr. Dana Petcu (PI)

Pagini web:

<https://dipet.hpc.uvt.ro/>

<https://dipet.eecs.qub.ac.uk/>

Contents

Obiectivele etape 2.....	3
Rezumatul Etapei 2: Prototip software.....	3
Gradul de atingere a rezultatelor estimate	4
Descriere științifică și tehnică	5
<i>WP1 – Streaming Algorithms with Tunable Precision</i>	<i>5</i>
T1.3 Transprecise streaming operators with numerical precision adaptation	6
T1.4 Transprecise streaming operators with algorithmic choice.....	7
T1.5 Mechanisms for tuning transprecise algorithms and analysis.....	7
<i>WP2 Programming Model and Runtime Implementation</i>	<i>12</i>
T2.2 Scheduling of transprecise Flink dataflow computations	13
T2.3 Library of Flink transprecise streaming operators	13
<i>WP3 Dynamic Resource Mapping and Middleware</i>	<i>14</i>
T3.1 Stream-processing application monitoring.....	14
T3.2 Performance modeling of transprecise stream processing	16
T3.3 Transprecise-aware dynamic resource mapping	17
<i>WP4 Deployment and Evaluation</i>	<i>17</i>
T4.3 Experimental evaluation and performance assessment.....	17
Concluzii.....	18
Bibliografie	18
Diseminare.....	19

Obiectivele etape 2

Proiectul DiPET investighează maparea dinamică și transparentă a aplicațiilor de procesare a fluxurilor de date în medii de calcul de tip Fog și Edge și folosește calculul cu trans-precizie cu scopul final de a îmbunătăți aspectele operaționale în termeni de utilizare a resurselor și consum de energie și de a îmbunătăți experiența utilizatorului.

Obiectivele principale a etapei 2 sunt fost:

- Identificarea unor metode/operatori peste care se pot optimiza folosind metode transprecise. În principal metode ML nesupervizate.
- Crearea unui set de date pentru testare și validare
- Executarea unor experimente folosind metode transprecise
- Instalarea și configurarea unui cluster folosind hardware cu resurse de calcul limitate similar cu dispozitivele din sisteme de tip Edge și Fog
- Extinderea platformei EDE pe baza rezultatelor obținute din experimente și conectarea acesteia mediului de dezvoltare.

Această abordare va crea noi oportunități de îmbunătățire a mai multor aspecte ale sistemelor de tip Fog și Edge, precum creșterea eficienței energetice și a performanței acestor sisteme (se vizează o îmbunătățire a performanței de 2 ori prin trans-precizie și o îmbunătățire a energiei similare). Mai mult, DiPET își propune să simplifice radical programarea, configurarea și desfășurarea procesării fluxului, permițând programatorilor să declare restricții de plasare la resursele dinamice, controlul de planificare și trans-precizie și / sau aproximarea, toate într-o singură specificație concisă. Planificatorul va fi unul în premieră, întrucât în prezent nu există un planificator bun pentru procesarea fluxurilor de date în medii de tip Fog, cu atât mai puțin conștient de trans-precizie, și va oferi transparență de funcționare deopotrivă pentru dezvoltatorii de aplicații și utilizatori. Sistemul DiPET va fi partajat comunității sub licențe open source și distribuit ca software reutilizabil, prin pachete software instalabile. În plus, vor fi dezvoltate cazuri de utilizare, alese dintre domeniile de aplicație relevante, centrate pe utilizator: detectarea intruziunilor în rețea și analiza video distribuită.

Rezumatul Etapei 2: Prototip software

Accentul principal a fost pe munca depusă în pachetul de lucru (WP1) unde echipa UVT este investigatorul principal. Acest pachet de lucru are sarcina de a crea/implementa algoritmi de streaming cu precizie reglabilă. În timpul primei etape, ne-am concentrat pe definirea calității, precum și a valorilor de performanță nefuncționale pentru transprecizie în contextul dispozitivelor Edge/Fog. Aceste cerințe stau la baza activității desfășurate pe parcursul acestei etape. Și anume definirea mai multor componente din Platforma DiPET.

Componenta principală dezvoltată în această etapă a fost Motorul de detectare a evenimentelor (EDE). Dezvoltarea inițială a EDE a fost realizată în cadrul proiectului de cercetare H2020 ASPIDE. Îmbunătățirea acestui cadru de bază a fost făcută pentru a include metode de optimizare transparente. Una dintre deciziile de proiectare luate în timpul acestei etape a fost extinderea EDE cu mai multe metode nesupravegheate de ultimă generație, concepute pentru detectarea anomaliilor.

Scopul principal a fost în identificarea unor operatori de analiză și predicție care pot fi optimizați folosind metode transprecise. Acești operatori sunt necesari pentru dezvoltarea unor strategii de distribuție care îmbunătățesc performanța aplicațiilor prin utilizarea calculului cu trans-precizie pentru a maximiza adaptabilitatea aplicațiilor la resursele disponibile în sisteme Fog și Edge.

Pentru a identifica acești operatori echipa UVT a colaborat cu partenerii din WP4 pentru a dezvolta diferite cazuri de utilizare. Rezultatul principal a acestei colaborări a fost realizarea unui set de date utilizat pentru experimente legate de optimizarea folosind metode transprecise și extragerea de candidați pentru integrarea în WP2 și modelarea în WP3.

Plăcile Nvidia Jetson Nano au fost adăugate infrastructurii existente UVT. Acestea vor fi utilizate în principal pentru micro-benchmarking al metodelor de învățare automată (ML) și pentru dezvoltarea și testarea clusterului Kubernetes utilizat pentru simularea cazurilor de utilizare Edge/Fog. Platforma de monitorizare descrisă în etapa 1 a fost și ea instalată împreună cu o instanță a platformei EDE. Experimentele și cazurile de utilizare vor fi folosite pentru a crea prima versiune a soluției DIPET pe acest cluster la începutul etapei 3.

Datorită pandemiei, întâlnirile partenerilor din proiect s-au desfășurat exclusiv on-line. Acestea au fost organizate de către coordonatorul din Belfast a proiectului în mod regulat, în fiecare miercuri, la ora 16:00, ora României prin Skype, în toată perioada raportată. Întâlnirile au avut rolul de a alinia activitățile echipelor partenere, a schimba idei și a analiza împreună direcțiile concrete, cu aspecte tehnice asociate, pentru activitatea de cercetare-dezvoltare.

Costurile de personal sunt conforme cu efortul depus și estimările inițiale, cu o depășire procentual mică. Activitățile cu caracter colaborativ (categoria deplasări), precum întâlnirile față-în-față și participările la conferințe internaționale (planificate în propunerea de proiect) au fost afectate și înlocuite cu întâlniri on-line și conferința internațională organizată la UVT. Aceste decizii sunt reflectate în abateri de la bugetul estimat al etapei (reducerea la zero a costurilor la deplasări și implicit a costurilor indirecte).

Gradul de atingere a rezultatelor estimate

Obiectivul științific și tehnic al etapei a fost îndeplinit în totalitate, cum vom prezenta în cele ce urmează. Este important de menționat faptul că partenerii responsabili pentru pachetele WP2 și WP3 au avut dificultăți în coordonarea activităților datorită situației create de pandemia COVID 19. Astfel, clusterul Jetson Nano (descriș în secțiunea aferentă WP3) a fost singurul mediu de testare accesibil echipei UVT.

Descriere științifică și tehnică

WP1 – Streaming Algorithms with Tunable Precision

UVT este coordonatorul a acestui WP astfel majoritatea efortului din etapa 2 a fost concentrat în acesta. Am extins EDE cu o gamă largă de caracteristici și metode adecvate pentru transprecizie. Arhitectura generală rămâne neschimbată, poate fi văzută în Figura 1. Mai multe metode de optimizare ghidată de hiper-parametri sunt, de asemenea, integrate în platformă. Acestea includ algoritmul genetic și soluțiile bazate pe metoda bayesiană.

Software-ul dezvoltat poate fi găsit la acest depozit Github:

<https://github.com/DIPET-UVT/EDE-Dipet>

Un prim draft cuprinzător al ghidului utilizatorului este, de asemenea, disponibil pe Github¹. Repository-urile auxiliare care conțin un cluster de testare Apache Flink și cod experimental suplimentar pot fi găsite și în rădăcina repository-ului UVT DIPET Github. Ar trebui să menționăm că mai multe repository-uri și ramuri de tip „work in progress” există și pe soluția UVT Gitlab. Toate experimentele din acestea vor fi îmbinate în repository-ul principal GitHub odată ce sunt finalizate.

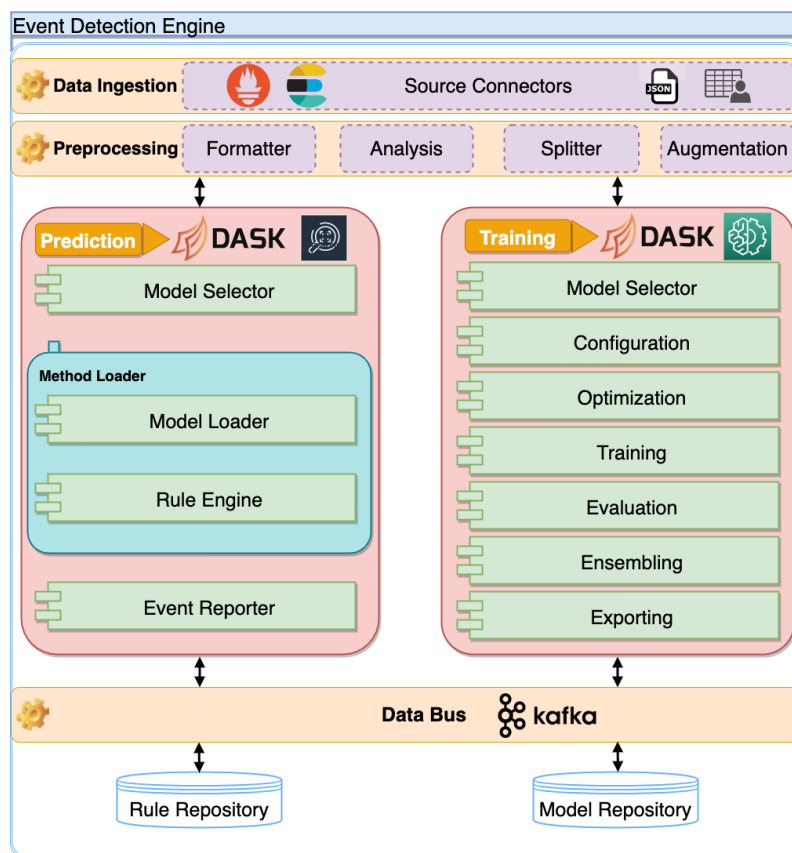


Figura 1 Arhitectura EDE

¹ <https://github.com/DIPET-UVT/EDE-Dipet/blob/master/README.md>

De menționat este faptul că îmbunătățirile aduse EDE acoperă și/sau sunt rezultatul muncii depuse pentru activitățile T1.2, T1.4 și T1.5.

T1.3 Transprecise streaming operators with numerical precision adaptation

În acesta activitate au fost identificați o serie de algoritmi de ML nesupervizați utilizabili pentru detecția de anomalii contextuale. În secțiunea de introducere am sugerat faptul că obiectivul nostru principal în timpul acestei etape a fost integrarea metodelor nesupravegheate de ultimă generație. Există multe motive pentru care acest lucru este de dorit. În primul rând, seturile de date disponibile pentru cercetare în DIPET sunt seturi de date din lumea reală, reprezintă sisteme și aplicații din lumea reală exploatate. Astfel, accesul la seturile de date etichetate este dificil. Seturi de date etichetate disponibile pentru evaluare comparativă și, eventual, chiar pentru scenarii de învățare de transfer, în cazul în care, totuși, modelele de predicție supravegheate rezultate vor avea, după toate probabilitățile, performanțe predictive slabe. În al doilea rând, metoda nesupravegheată poate fi utilizată în scenarii mai variate și chiar în sisteme care sunt exploatate activ. Numai acest fapt crește aplicabilitatea și utilitatea lor potențială. Din păcate, este probabil mai dificil de lucrat cu ele, iar validarea rezultatelor este destul de diferită decât în cazul metodelor supravegheate.

Am integrat în EDE o bibliotecă Python numită PyOD². Este conceput pentru detectarea valorii aberante (adică anomalii) în diferite scenarii. În prezent, integrăm toți cei 30 de algoritmi de ultimă generație. Cu toate acestea, am efectuat experimente doar cu un subset dintre acestea; Detector de valori aberante bazat pe unghi (ABOD), factor de valori aberante bazat pe clustere (CBLOF), detecție a valorii aberante pe baza de histogramă (HBOS), pădure de izolare, KNN cei mai apropiați vecini (KNN), factor de valori aberante locale (LOF), determinant de covarianță minimă (MCD), SVM cu o singură clasă (OCSVM), Combinație selectivă locală (LSCP), Codificare automată (AE), Codificare automată variațională (VAE), Învățare activă generativă adversară cu un singur obiectiv (SOGAAL), Învățare activă generativă adversară cu obiective multiple (MOGAAL).

De remarcate sunt metodele VAE, SOGAAL și MOGAAL. Acestea sunt metode de detectare a anomaliilor nesupravegheate bazate pe Deep Learning. Pentru a le integra cu succes în EDE, a trebuit să facem modificări substanțiale codului original. Ultimele două metode au doar modificări funcționale, în timp ce prima are unele îmbunătățiri suplimentare utilizate în timpul experimentelor de benchmarking și adaptare tranprecisă.

Valorile Shapely au fost introduse pentru prima dată în domeniul jocurilor de coaliție. A fost folosit pentru a determina contribuția fiecărui jucător pentru orice decizie dată. Această metodă este acum utilizată și în domeniul Explainable AI, în cazul metodelor ML nesupravegheate poate determina ce caracteristică dintr-un set de date a contribuit la etichetarea unui eveniment ca fiind anormal. Acest lucru are două implicații în cazul nostru.

În primul rând, poate ajuta la determinarea cauzei fundamentale a unui eveniment anormal. O sarcină dificil de realizat în cazul metodelor nesupravegheate. Și în al doilea rând, teoretic ne poate oferi o estimare bună a importanței globale a caracteristicilor care poate fi utilizată

² <https://pyod.readthedocs.io/en/latest/>

pentru selecția caracteristicilor. Un avertisment pentru utilizarea valorii Shapely este că necesită seturi de date etichetate.

Această problemă poate fi rezolvată prin antrenarea mai întâi a unei metode nesupravegheate pe întregul spațiu de caracteristici. Apoi, am folosit predicția modelului rezultat ca etichete și, în final, am redus spațiul caracteristicilor pe baza importanței globale a caracteristicilor calculate folosind valori Shapely. Detalii legate de rezultatele experimentelor sunt detaliate în T1.5.

Calculul valorilor Shapely sunt extrem de costisitoare în cazul spațiilor caracteristice mari. Din acest motiv, am integrat în EDE o bibliotecă AI explicabilă de înaltă performanță numită SHAP³ pe care o putem folosi pentru aceste calcule.

T1.4 Transprecise streaming operators with algorithmic choice

De acest task este responsabil partenerul Queen's University of Belfast (QUB). În momentul de față se lucrează asupra identificării diferitelor metode de tip Deep Learning pentru procesarea video. Până în momentul de față sau realizat două întâlniri via Skype pentru a identifica punctele de colaborare între UVT și QUB. Sa decis ca metodele folosite pentru optimizarea metodelor identificate în T1.3 și T1.4 se vor adapta pentru T1.4. Mai precis adăugarea de suport pentru metode de tip „out-of-core”. Prin includerea în T1.3 a metodelor AE, VAE, SOGAAL și MOGAAL această cerință a fost satisfăcută urmează testarea efectivă în etapa 3.

T1.5 Mechanisms for tuning transprecise algorithms and analysis

În această activitate am implementat diferite metode de optimizare și selecție de caracteristici (features) pentru a optimiza performanțele metodelor selectate în T1.4.

Pentru setul nostru inițial de experimente am ales 3 metode de detectare nesupravegheată; CBLOF, IsolationForest și VAE. Aceste 3 au fost alese deoarece sunt abordări radical diferite pentru detectarea anomaliilor. Datele folosite pentru experimente sunt cele rezalite în colaborare cu partenerul Universitat Politècnica de Catalunya (UPC). Detalii sublimentare legate de acest set de date se pot găsi la secțiunea aferentă WP4 din acest raport.

Primul pas a fost preprocesarea datelor de instruire și testare. Am eliminat caracteristicile cu variație scăzută din setul de date, deoarece acestea au un impact redus sau deloc asupra pregătirii modelului. Apoi am normalizat datele la intervalul [0,1]. Acest pas este necesar deoarece unele metode ML (în special VAE) au performanțe mai bune dacă datele sunt normalizate în acest fel.

Majoritatea metodelor de detectare a anomaliilor nesupravegheate și a valorii aberante necesită definirea unui factor de contaminare așteptat pentru instruire. Deoarece evenimentele anormale se întâmplă rar și dorim să evităm pe cât posibil falsele pozitive, setăm contaminarea la 0,005.

³ <https://shap.readthedocs.io/en/latest/>

Apoi am antrenat toate cele 3 metode pe setul de antrenament. Figura 3 prezintă o diagramă 3D a evenimentelor anormale detectate cu Isolation Forest. Ar trebui să menționăm că am folosit metoda PCA pentru reducerea dimensionalității doar pentru vizualizare, toate evenimentele anormale detectate marcate cu un X roșu sunt precise folosind întregul spațiu de caracteristici.

Toți cei 3 algoritmi au detectat un total de 37 de evenimente anormale în setul de antrenament de peste 8000 de evenimente. Acesta este un rezultat direct al parametrilor de contaminare setați în timpul antrenamentului. De asemenea, ar trebui să menționăm că am selectat în mod special un interval de timp în care presupunem că nu au avut loc evenimente anormale. Chiar și așa, nu numai că metodele ML au detectat același număr de anomalii, dar sunt în mare parte aceleași anomalii între toate metodele. Cea mai mare diferență este între Pădurea de Izolație și VAE, unde un total de 7 anomalii sunt diferite.

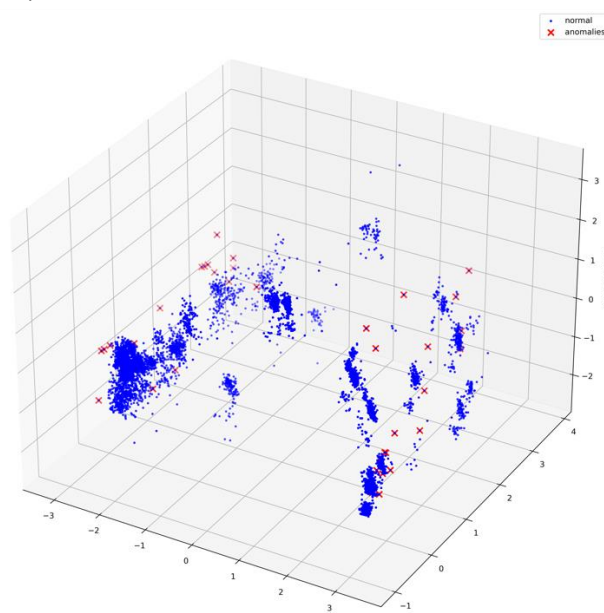
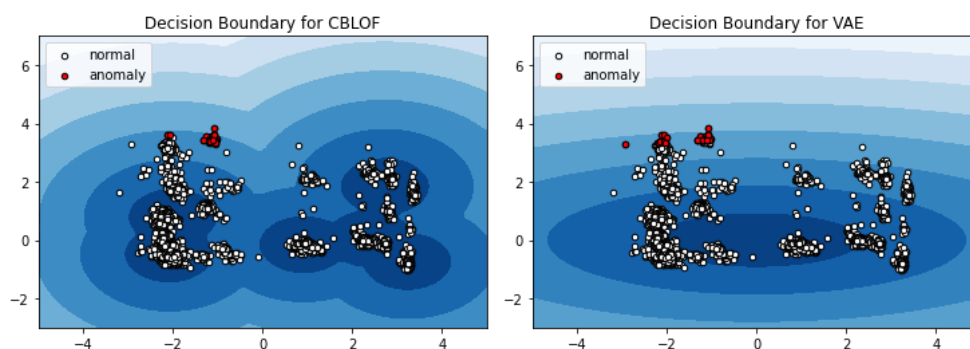


Figura 2 Datele de antrenare după 3 componente PCA cu anomalii



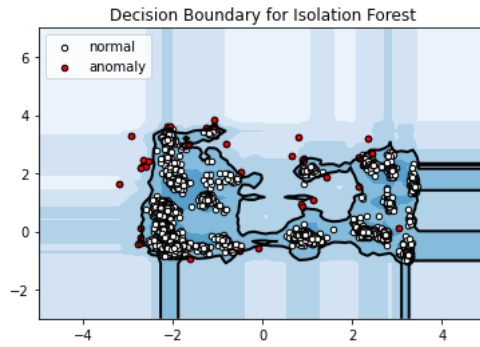


Figura 3 Limitele de decizie calculate pentru toate cele 3 metode

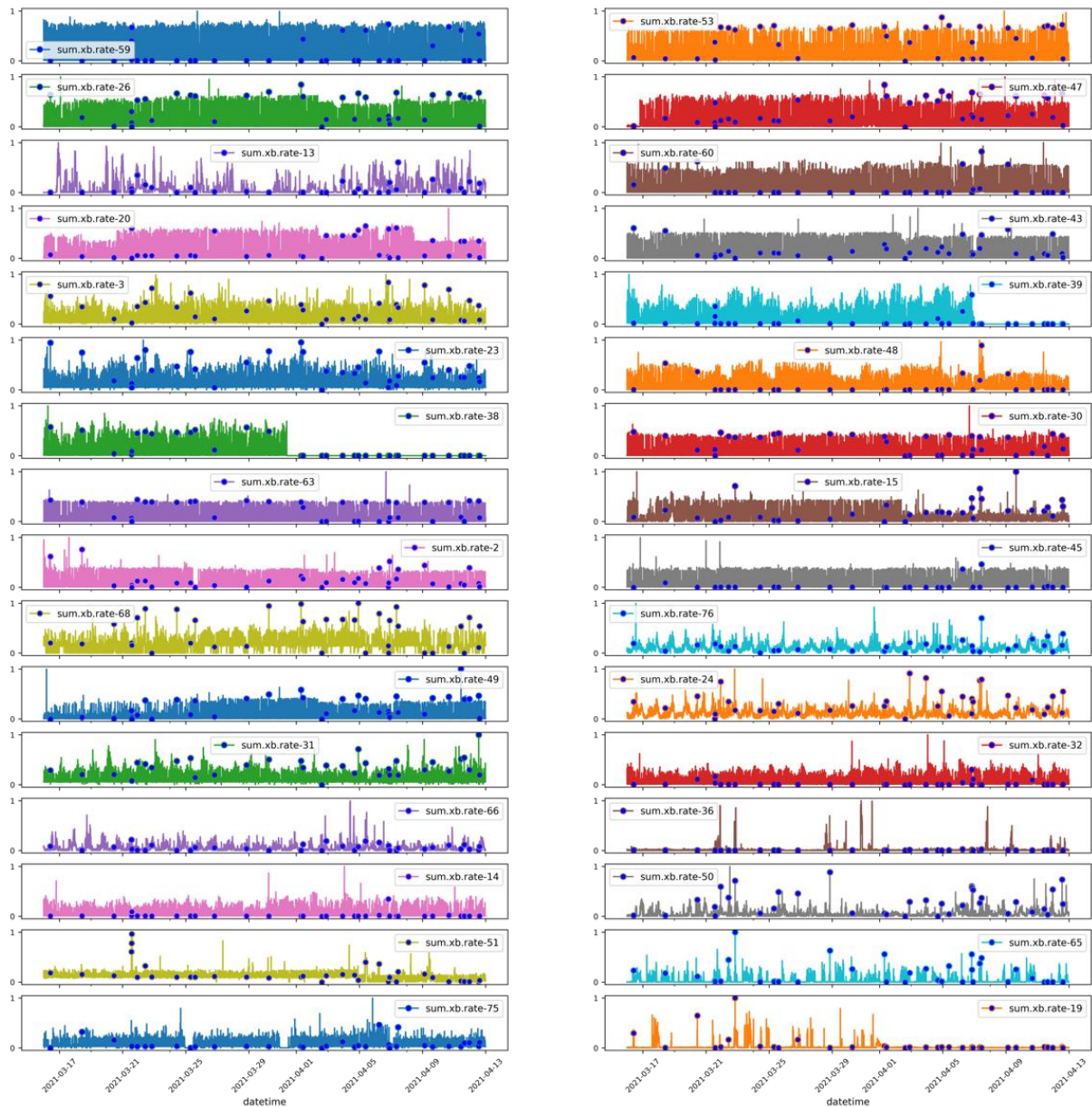


Figura 4 Anomaliile detectate pe setul de antrenament

Figura 5 arată limita de decizie calculată pentru toate cele 3 metode. Putem vedea cu ușurință diferențele dintre modul în care fiecare metodă ML „decide” ce este un eveniment anormal și ce nu. Un interes deosebit este limita de decizie calculată pentru IsolationForest. Deoarece această metodă se bazează pe aceleași concepte de bază ca modelele de arbore de decizie supravegheat (în special RandomForest), putem presupune cu siguranță că unele dintre idiosincrazii pot fi găsite și în IsolationForest. Figura 3 sugerează cu tărie că supraadaptarea modelului poate fi o problemă. Acest lucru poate duce la o performanță slabă a eșantionului.



Figura 5 Anomaliile detectate pe setul de testare

Figurile 4 și 5 arată anomaliile detectate pe 32 de caracteristici din seturile de date de antrenament și de testare. Putem vedea că în cazul setului de testare evenimentele anormale sunt aparent plasate aleatoriu. Cu toate acestea, după cum sa menționat anterior, suprapunerea dintre metodele 3 ML sugerează că au existat unele evenimente anormale. Aceste evenimente pot fi explicate prin faptul că rețeaua mesh Guifi Net este utilizată în mod activ, astfel încât sunt de așteptat creșteri aleatorii ale utilizării rețelei.

Figura 6 este extrem de interesantă prin faptul că arată o grupare mare de evenimente anormale exact în punctul în care știm că nodul ISP a eșuat. Dar în perspectivă, datele de testare au 800 de evenimente și cele 3 metode de predicție sunt după cum urmează: CBLOF 125, IsolationForest 74, VAE 134. Putem vedea că numărul de evenimente anormale detectate depășește cu mult factorul de contaminare setat în timpul antrenamentului.

Următorul pas a fost să calculăm valorile Shapely folosind SHAP. Figura 6 arată explicația pentru un eveniment anormal. Partea stângă a figurii arată acele caracteristici care „împing” predicția către un scor mai mare de anomalie, în timp ce cele albastre „o trag” înapoi. De asemenea, putem vedea contribuția fiecărei valori caracteristice pentru o predicție.

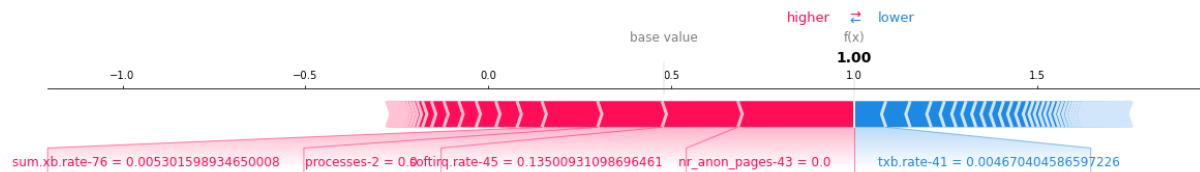


Figura 6 SHAP Force Plot

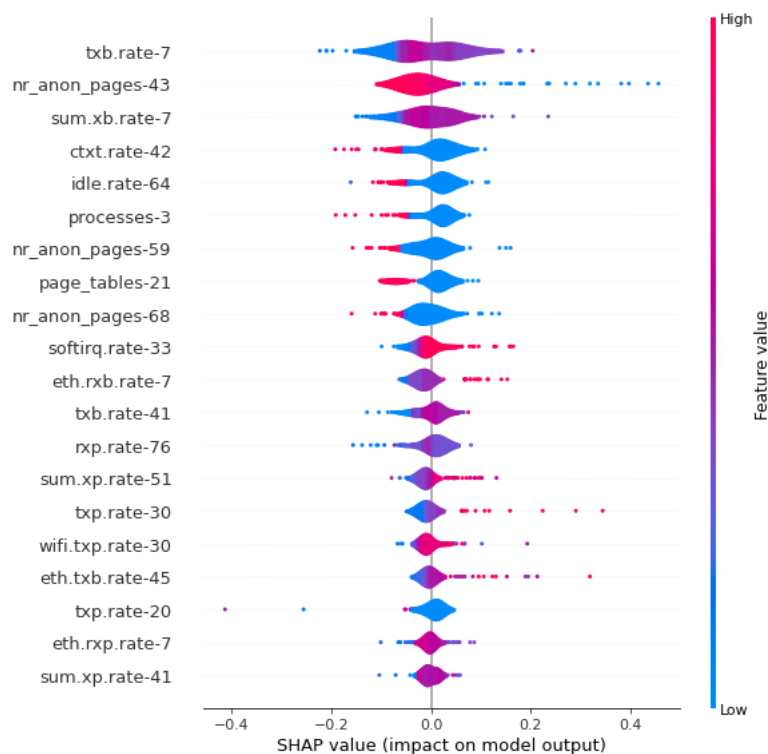


Figura 7 SHAP Violin plot, feature importance

Figura 7 prezintă o vedere a importanței caracteristicilor globale calculate pe baza valorilor SHAP. Folosind aceasta, am calculat că 31 de caracteristici reprezintă peste 90% din informațiile necesare pentru detectarea evenimentului nostru anormal.

Pentru a doua fază a experimentelor am folosit doar caracteristica 31, bazată pe importanța caracteristicii valorii Shapely. Restul experimentului a fost nemodificat. Având în vedere că spațiul original de caracteristici a fost cu mult peste 1500 de caracteristici, performanța metodelor ML cu doar 31 de caracteristici a fost remarcabilă. Cele 3 metode au detectat următoarele evenimente anormale.: CBLOF 129, IsolationForest 117, VAE 127. Cele două metode anterioare au avut rezultate mult mai bune, IsolationForest aproape dublând evenimentul anormal detectat.

Acest lucru poate fi explicat prin modul în care funcționează IsolationForest. Fiind o metodă de ansamblu alcătuită din arbori de izolare pe care o construiește prin selectarea aleatorie a caracteristicilor. Cu cât există mai multe caracteristici, cu atât este mai mică probabilitatea de a selecta combinații eficiente de caracteristici pentru construcția arborelui de izolare. Prin eliminarea caracteristicilor mai puțin pertinente, creștem foarte mult șansa de a selecta combinații bune de caracteristici. Timpii de antrenament și inferență au fost, de asemenea, reduse drastic, în unele cazuri de peste 10 ori.

Apoi am încercat alte metode de selectare a caracteristicilor descrise în acest raport în secțiunea T2.3. Rezultatele nu au fost la fel de bune ca la cele obținute folosind Shapely Values. Reducerea spațiului pentru funcții a variat de la 32 la 100 de caracteristici, în timp ce rata de detectare pentru setul de antrenament a scăzut la o singură cifră. Interesant este că rata de detectare a celor 37 de evenimente anormale din setul de antrenament a rămas în mare parte neschimbată. O altă observație interesantă este că, deși am obținut doar instanțe anormale cu o singură cifră, toate au fost în intervalul de timp al eșecului nodului ISP cunoscut. Acest lucru poate indica mai multe lucruri. Poate însemna că limităm posibilitatea detectării fals pozitive, poate indica, de asemenea, că prin eliminarea unor caracteristici eliminăm în mod esențial capacitatea de a detecta toate evenimentele anormale din setul de testare. În esență, trebuie să stabilim câte tipuri de evenimente anormale au avut loc atunci când nodul ISP a eșuat.

Rezultatele prezentate aici sunt doar o parte din rezultatele obținute pe parcursul acestei etape. O lucrare de jurnal completă care detaliază toate experimentele și rezultatele este în prezent în lucru și este programată să fie publicată într-un Jurnal Open Access până la începutul anului 2022. În momentul scrierii acestui raport lucrarea este în proporție de 90% completă.

Următorul pas va fi includerea experimentelor pentru metode supervizate folosind metode de optimizare de hiper-parametri. De asemenea este important să menționăm faptul că setul de date folosit în aceste experimente prezintă similitudini cu problemele din WP3 astfel obiectivele etapei 3 vor fi în principal legate de integrarea rezultatelor obținute în această activitate în WP3.

WP2 Programming Model and Runtime Implementation

În cadrul acestui WP, echipa UVT are rol de suport. Scopul principal din punctul de vedere a echipei UVT este de a identifica și implementa operatori speciali identificați din WP1 în

extensii Apache Flink și măsurarea unor metrici care indică performanțele metodelor identificate.

T2.2 Scheduling of transprecise Flink dataflow computations

În urma a mai multor discuții cu principalul partener din acest WP, Foundation for Research and Technology – Hellas (FORTH) sa decis ca partenrii să țină cont de anumite metrici de performanță din timpul antrenării și validării metodelor de procesare. Aceste metrici vor servi ca bază a metodelor de planificare implementate de către FORTH. Echipa UVT a colectat o serie de măsurători (timp de antrenarea/inferența, debitul fluxul de date maxim, latența instanțierii metodelor predictive antrenate etc.) în momentul execuției experimentelor din WP1. Urmează ca în etapa 3 acestea să fie folosite în mod direct de către soluția de planificare (scheduling).

T2.3 Library of Flink transprecise streaming operators

În aceasta activitate au fost identificate o serie de operatori care pot fi folosiți în mod dinamic în timp real pentru reducerea cantității de date pentru diversele metode de procesare identificate în DIPET. O metodă de optimizare precisă este aceea de selecție a caracteristicilor. Acest lucru permite o reducere a spațiului caracteristicilor. Cel mai bun scenariu este că prin reducerea caracteristicilor putem crește efectiv performanța predictivă, reducând în același timp atât timpii de antrenament și de inferență, cât și cerințele de calcul. Din păcate, un rezultat tipic este un compromis posibil între cerințele de calcul și performanța predictivă.

Am identificat și implementat mai multe metode de selectare a caracteristicilor nesupravegheate. Aceste metode vor fi ulterior transformate în operatori transprecise Flink în această activitate. Printre metodele de selectare a caracteristicilor avem:

- *Măsura Kurtosis*– Această măsură poate fi calculată pentru fiecare caracteristică ca o formă de metodă de secțiune a caracteristicilor nesupravegheate. Măsura K a unui set de caracteristici notate ca $f_1 \dots f_N$ este calculată prin standardizarea mai întâi a fiecărei caracteristici la $k_1 \dots k_N$ cu o medie zero și o abatere standard unitară:

$$K(k_1 \dots k_N) = \frac{\sum_{i=1}^N k_i^4}{N}$$

Acest lucru ne permite să descriem distribuția de probabilitate a datelor. Un K ridicat poate indica un grad ridicat de abatere într-o caracteristică.

- *Diferența medie absolută* – Este folosită pentru a calcula diferența absolută față de valoarea medie. Indică un potențial discriminatoriu mai mare. Se calculează astfel:

$$MAD_i = \frac{1}{n} \sum_{j=1}^n |X_{ij} - \bar{X}_i|$$

unde \bar{X}_i se calculează astfel:

$$\bar{X}_i = \left(\frac{1}{n}\right) \sum_{j=1}^n X_{ij}$$

- **Corelare** - poate fi folosit și ca o formă de metodă de selecție a caracteristicilor nesupravegheate. O corelație ridicată între caracteristici indică același efect asupra variabilelor dependente. Prin calcularea valorii p putem măsura efectul pe care o caracteristică îl are asupra întregului set de date. De obicei, o corelație ridicată între caracteristici va duce la eliminarea caracteristicii. În cazul nostru am folosit corelația Pearson, totuși, pot fi utilizate și clasamentele Spearman și Shapiro. Figura 8 arată corelația Pearson ca hartă termică pentru setul de date de antrenament.

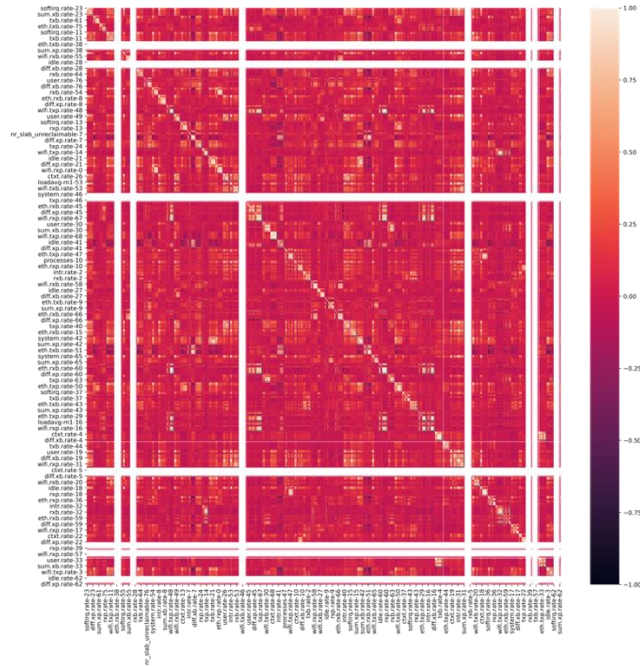


Figura 8 Corelarea Pearson pentru setul de date de antrenare

Metoda bazată pe valori Shapely descrisă în secțiunea aferentă activității T1.3 nu este implementabilă ca și operator în această etapă de oare ce calcularea acesteia reprezintă un efort considerabil în contextul sistemelor Edge/Fog.

WP3 Dynamic Resource Mapping and Middleware

Având în vedere că EDE este utilizat atât pentru unele cazuri de utilizare, cât și ca componentă în platforma finală DIPET în virtutea faptului că sunt probleme similare cu cerințe funcționale și nefuncționale similare. Rezultatele experimentelor prezentate din secțiunea aferentă WP1 Sunt pertinente parțial pentru WP3. De asemenea, platforma de prototipare prezentată în raportul pentru etapa 1 a fost extins și integrat în infrastructura de calcul a UVT. Permițând monitorizarea și execuția soluțiilor dezvoltate în cadrul DIPET

T3.1 Stream-processing application monitoring

În perioada actuală de raportare, s-a instalat clusterul de dispozitive mini-EDGE personalizat (achiziționat din proiect în prima etapă), care constă din 10 carduri Nvidia Jetson Nano⁴ și o stocare SSD dedicat USB3.0 (9). Nodurile Nvidia Jetson au următoarele specificații:

⁴ NVIDIA Jetson Nano - <https://developer.nvidia.com/embedded/jetson-nano-developer-kit>

- **CPU:** Quad-Core ARM A57 @ 1.43 GHz;
- **GPU:** 128-core Maxwell;
- **Memory:** 4GB 64-bit LPDDR4;
- **Storage:** microSD and 256GB SATA3 USB3.0 SSD;
- **Connectivity:** 1Gbps Ethernet;
- **USB:** 4x USB3.0, 1x USB 2.0 Micro-B;
- **Other capabilities:** video encode: 4K @ 30 | 4x 1080p @ 30 | 9x 720p @ 30 (H.264/H.265), video decode: 4K @ 60 | 2x 4K @ 30 | 8x 1080p @ 30 | 18x 720p @ 30 (H.264/H.265).

Acestea sunt dispozitive mai puțin pline decât serverele de calcul clasice, dar puterea lor de calcul hibridă CPU/GPU încorporată este capabilă să suporte mai multe aplicații de rețea neuronală pentru a rula în paralel pentru scenarii care implică tehnici și algoritmi de învățare automată (ML). Mini-clusterul din UVT va fi folosit în principal pentru a efectua experimente în domeniul calculului cu transprecizie în ML.

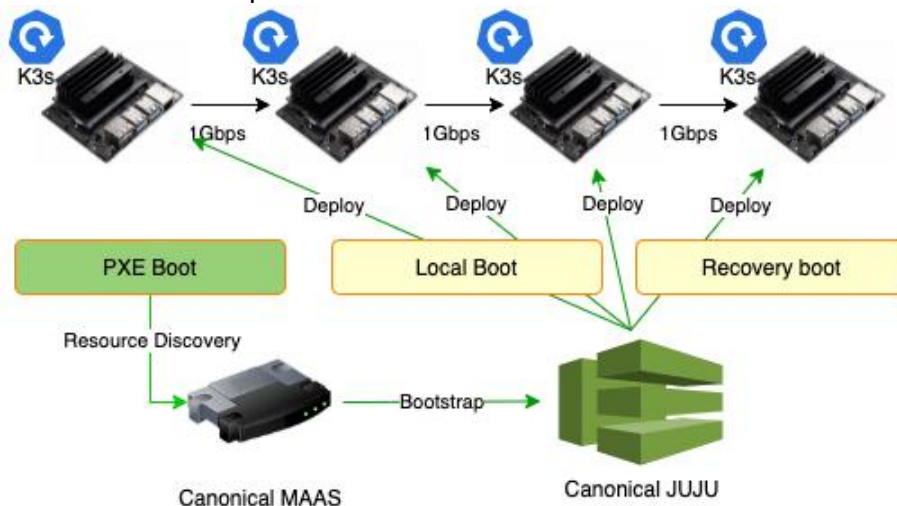


Figura 9: Managementul Clusterului Edge

Aceste dispozitive pornesc în mod normal de pe un card SD atașat unde este instalat sistemul de operare. Pentru a crește spațiul de stocare, viteza și persistența, am adăugat suport pentru pornirea sistemului de operare de pe hard disk SSD atașat USB3.0. Fiecare dispozitiv Nvidia Jetson Nano este configurat să pornească de pe hard diskul SSD și are o conexiune externă de 1 Gbps pentru transferul de date și operațiuni de bootstrap. Spațiul de pe hard diskul SSD este împărțit pe 3 partiții: (1) boot, unde se află kernel-urile și discurile ram, (2) sistemul de fișiere rădăcină, spațiul real al utilizatorului sistemului de operare și (3) partiția de recuperare, utilizată pentru readucerea dispozitivului la o stare predefinită în caz de defecțiuni. Procesul de pornire folosește doi pași: (a) inițial, se execută o pornire PXE de rețea pentru pornirea unui sistem de operare transmis prin rețea și, dacă nu sunt primite informații despre rețea, se inițiază (b) boot-ul local, folosind partiția de pornire și ulterior sistemul de operare instalat local pe hard diskul SSD. Pornirea PXE în rețea este utilizată pentru a susține bootstrapping automat al dispozitivului folosind un serviciu de furnizare bare metal, Canonical MAAS⁵. Acest serviciu va identifica resursa Edge și va porni sistemul de operare inițial. În plus, folosind

⁵ Canonical MAAS - <https://maas.io/>

Canonical JUJU⁶ mediul de rulare de pe nodul Edge va fi personalizat prin implementarea stivei software necesare. Pentru DiPET, o versiune ușoară a Kubernetes, K3s⁷, va fi utilizată ca orchestrator pentru diferite scenarii de implementare a software-ului care urmează să fie testate pe resursele de vârf oferite de cluster. Folosind configurația curentă, mini-clusterul poate fi personalizat pentru a găzdui diverse medii de testare și dezvoltare asemănătoare Edge.



Figura 10: Nodurile Clusterului UVT mini-EDGE

Soluții de monitorizare prezentată inițială bazată pe Prometheus⁸ în raportul aferent etapei 1 cât și în livrabilul D3.1 a fost instalată pe clusterul Jetson Nano prezentat.

T3.2 Performance modeling of transprecise stream processing

Echipa UVT împreună cu partenerul principal responsabil pentru această activitate, Institute de Recherche Informatique et Systemes Aleatoires (IRISA) a colaborat în mod activ în etapa 2. Experimentele realizate în cadrul WP1 folosesc metode ML utilizabile și pentru modelarea

⁶ Canonical JUJU - <https://jaas.ai/>

⁷ K3s Lightweight Kubernetes - <https://k3s.io/>

⁸ <https://prometheus.io/>

proceselor. Metodele descrise în livrabilul D3.1 dezvoltate de către IRISA vor fi integrate în platforma EDE la începutul etapei următoare. Aceste metode sunt deja implementate conform cerințelor EDE, testarea integrării nu a fost realizată până în momentul scrierii acestui raport.

T3.3 Transprecise-aware dynamic resource mapping

Predicțiile metodelor obținute folosind metodele din T3.2 vor fi folosite în procesul de mapare a resurselor din soluția DIPET. Interfețele de integrare au fost deja realizate. Soluția EDE, dezvoltată de către echipa UVT fiind capabilă să monitorizeze toate metricile pertinente pentru modelarea unei aplicații de streaming bazate pe operatori „transprecise”. În etapa 3 focusul principal va fi pe realizarea unor experimente asupra cazurilor de utilizare care folosesc soluția DIPET și sunt executate pe mediul de execuție prezentat în T3.1.

WP4 Deployment and Evaluation

În acest pachet de lucru UVT are rol de suport pentru partenerii responsabili cu definirea și implementarea cazurilor de utilizare. Echipa UVT a avut o colaborare extrem de importantă cu Universitat Politècnica de Catalunya (UPC), realizând un set de date utilizat în experimentele prezentate în secțiunea aferentă WP 1.

T4.3 Experimental evaluation and performance assessment

În raportul etapei anterioare am descris cazul de utilizare Guifi Net⁹ și setul de date disponibil. Din motive de concizie, nu vom descrie din nou acest set de date în acest raport. În timpul ultimului reper, UVT și-a continuat colaborarea cu Universitat Politècnica de Catalunya (UPC), am reușit să extindem setul de date cu diferite funcții noi. Din punctul de vedere al proiectului, am identificat mai multe evenimente reale live pe care le putem folosi pentru experimentele noastre. Unul dintre cele mai importante evenimente a fost o defecțiune a nodului ISP care a avut loc într-o perioadă de 6 ore în aprilie 2021. Datele colectate de la acest eveniment conțin valori de la toate nodurile afectate, aproximativ 60 (unele noduri unde noduri experimentale, comportamentul lor este dificil de conectat cu întreruperea ISP-ului).

Am folosit acest eveniment pentru cea mai mare parte a muncii noastre experimentale în timpul acestei etape. Datele sunt împărțite într-un set de antrenament care conține date de monitorizare în valoare de aproximativ 1 săptămână (8000 de evenimente cu 1600 de metrici monitorizate pentru fiecare eveniment) și un set de testare care conține aproximativ 800 de evenimente cu aceeași caracteristică dimensională ca setul de antrenament. Setul de date ulterior conține evenimentul de eroare ISP. Setul de date obținut va fi publicat pe platforma Zenodo¹⁰ la începutul etapei 3.

⁹ <https://guifi.net/>

¹⁰ <https://zenodo.org/>

Concluzii

Obiectivele etapei au fost îndeplinite în totalitate. Software-ul dezvoltat este utilizat ca parte integrantă alături de codurile partenerii străini ai proiectului. Calitatea și utilitatea acestor prototipuri proof-of-concept a fost testată în condiții reale. Întâlnirile periodice ale partenerilor au fost productive și esențiale pentru avansul și sincronizarea activităților echipelor.

Concluziile pertinente per pachet de lucru sunt următoarele:

- **WP1** – Echipa UVT a realizat o serie de experimente unde au fost identificate o serie de metode bazate pe ML cât și unele metode de optimizare transprecise. Analiza amănunțită a problematicii identificate în cazurile de utilizare a scos la arătat faptul că acestea au o suprapunere cu o parte din problematicile identificate în WP2 și WP3. Astfel rezultatele experimentelor au dus la integrate în platforma EDE a mai multor metode ML cât și a unor metode de optimizare. Pentru etapa 3 se dorește extinderea experimentelor prin folosire de metode mai avansate de optimizare transprecisa.
- **WP2** – Pe baza experimentelor din WP1 echipa UVT a realizat o serie de măsurători utilizate pentru implementarea metodelor de planificare. De asemenea, au fost identificate și câțiva operatori care urmează a fi implementați în Apache Flink în cadrul etapie 3. Alți operatori urmează să fie identificați.
- **WP3** - Echipa UVT a instalat Cluster-ul format din Jetson Nano peste care a instalat soluția de monitorizare propusă în etapa 1. Îmbunătățirile aduse platformei EDE precum și metodele ML identificate în WP1, permit modelarea aplicațiilor de transprecise de streaming. Colaborarea cu partenerii din WP3 a dus și la integrarea în EDE a mai multor metode de modelare dezvoltate. Validarea acestor noi metode urmează a fi realizată în etapa 3.
- **WP4** – Setul de date obținut din colaborarea UVT cu UPC a fost folosit pentru experimentele din WP1. Acest set de date este singurul dezvoltat în cadrul DIPET pentru momentul de față. Publicarea a acestui set de date va fi realizat la începutul etapei 3.

Prototipurile software de tip proof-of-concept sunt disponibile public în GitHub la adresa <https://github.com/DIPET-UVT/EDE-Dipet>

Bibliografie

Rezultatele de mai sus au fost cuprinse partial în livrabilele D1.1 și D3.1

- D1.1: Quality and Non-Functional Performance Metrics for Edge/Fog Computing (Disponibil în EVoC)
- D3.1: Initial Monitoring Framework and Performance Model (Disponibil în EVoC)
- D4.1: Testbed preparation, requirements and initial experiments

care includ starea de fapt cu referințe la peste 50 de lucrări.

Diseminare

Rezultatele activităților de diseminare ale proiectului, inclusiv ale echipei UVT, sunt disponibile la <https://dipet.eeecs.qub.ac.uk/dissemination/#publications>

Nu au fost efectuate deplasări ale echipei UVT din motive de pandemie.

1. Software

Principalul rezultat al etapei 2 „Prototip software” este disponibil în regim open-access la <https://github.com/DIPET-UVT/EDE-Dipet>

2. Livrabile

Proiectul DIPET este un proiect european colaborativ cu livrabile care sunt prezentate la finalul anului de proiect (Martie a anului curent) organizatorilor competiției CHIST-ERA. Livrabilele din martie 2021 la care echipa UVT a contribuit (și disponibile în platforma EVOC, ca anexe la raport) sunt:

- Livrabilul D1.1: Quality and Non-Functional Performance Metrics for Edge/Fog Computing
- Livrabilul D3.1: Initial Monitoring Framework and Performance Model

3. Articole directe

O lucrare care cuprinde o parte din rezultatele proiectului a fost prezentată la workshopul JCSd din Barcelona în 2-3 Sept 2021:

- Llorenç Cerda-Alabern, *Gabriel Iuhasz*, Anomaly Detection in Wireless Community Networks using PCA, 2-3 Sept 2021, Jornadas de Concurrencia y Sistemas Distribuidos 2020+1 (accesibilă la <http://tomir.ac.upc.edu/node/932>)

Programul evenimentului este disponibil la <https://jcsd2020.ac.upc.edu/sample-page/>

4. Indirecte

Contribuțiile tuturor echipelor din proiectul CHIST-ERA au fost prezentate de către coordonatorul din Belfast în Aprilie 13 la CHIST-ERA projects seminar 2021 în 12-14 Aprilie 2021. Programul este disponibil la <https://www.chistera.eu/projects-seminar-2021-programme>

O lucrare care a fost realizată folosind Clusterul DIPET EDGE de colegi din departamentul care implementează DIPET este:

- Orhei C, Bogdan V, Bonchis C, Vasiliu R. Dilated Filters for Edge-Detection Algorithms. *Applied Sciences*. 2021; 11(22):10716. doi: 10.3390/app112210716

Revista Applied Science este revistă clasificată „ISI galben” (în primele 50% conform din categorie în clasamentul întocmit de Clarivate Analytics pentru 2020).

O altă lucrare care a fost realizată folosind clusterul DIPET EDGE de către doi doctoranzi este:

- Kristijan Cincar, Todor Ivascu, Rule-based health status evaluation system, IEEE International Conference on e-Health and Bioengineering, Iași, 18-19 Noi 2021

Doi dintre membrii proiectului DIPET (D. Petcu și S. Panica) sunt persoane cheie (<https://synasc.ro/2021/committees/>) în organizarea conferinței internaționale SYNASC 2021 (<https://synasc.ro/2021/>), cu publicație proceeding în IEEE CSP (așteptat să fie indexat ISI proceedings). Programul evenimentului organizat este disponibil la: <https://easychair.org/smart-program/SYNASC2021/>. În mod special track-ul de Distributed Computing a fost aliniat cu tematicile DIPET (<https://synasc.ro/2021/distributed-computing/>). Trei dintre lucrările acceptate spre publicare sunt elaborate de doctoranzii directorului local al DIPET și au folosit clusterul DIPET EDGE:

- Sorin Valcan, Convolutional Neural Network Training System For Eye Location On Infrared Driver Recordings Using Automatically Generated Ground Truth Data
- Alexandru Ionascu, Extracting football players video sprites from broadcast video
- Sebastian-Aurelian Stefaniga, Methods of acceleration for feature extractions in medical imaging using GPU processing

PI / UVT

Prof. Dr. Dana Petcu

