România
Ministerul Apărării Naționale
Academia Tehnică Militară "*Ferdinand I*"

Facultatea de Sisteme Informatice și Securitate Cibernetică
CALCULATOARE ŞI SISTEME INFORMATICE PENTRU APĂRARE ŞI SECURITATE NAŢIONALĂ

Platformă de Analiză Automată a Atacurilor Data Poisoning asupra unei Infrastructuri de Învățare Federate

Coordonator Științific
Cpt. conf. dr. ing. Iulian Aciobăniței

Absolvent
Sd. Sg. Maj. Lepădatu Tudor

București

An

Mulțumiri pentru persoanele care au sprijinit procesul de realizare a acestei lucrări

Referatul Coordonatorului Științific

Referatul reprezintă un text în care coordonatorul științific sumarizează, ulterior finalizării conținutului efectiv, efortul pe care l-ați depus și trage concluzii cu privire la gradul de realizare a obiectivelor propuse inițial (cele prezentate în cadrul detalierii). De regulă, acest referat nu depășește cele 4 pagini alocate în acest document.

Tema Proiectului de Diplomă

Tema proiectului de diplomă (sau detalierea) reprezintă un text realizat de coordonatorul științific, în lunile următoare propunerii titlului proiectului de diplomă către facultate, în care detaliază nevoia reală a implementării unui astfel de proiect și modul în care se dorește a fi implementat. În plus, poate prezenta structura pe capitole a viitoarei lucrări, anexele ce vor fi incluse și sursele bibliografice din care studentul se va informa. De regulă, acestă detaliere nu depășește cele 4 pagini alocate în acest document.

Abstract

The Artificial Intelligence integration with tools and applications has evolved since 2020s and the cybersecurity scene tries to adapt frequently. Information security and data integrity is more important than never before, being used by Machine Learning models or Neural Networks trained to perform specific tasks.

From a data science point of view, the quality of information is much important than securing it. The AI evolution has led to the creation of different attack boundaries, from changing the model parameters to perform data poisoning schemes. Confidentiality is the key in maintaining unique aspects for each entity involved in the federated learning process.

In this paper, data poisoning attacks are studied with different options in a segregated simulated infrastructure called federated learning, in which each client may change its scope intentionally or unintentionally. Each simulation has its own configuration providing data scientist with a dedicated environment for testing its machine learning algorithm againts data poisoning attacks.

Cuprins

Listă de figuri

Listă de Abrevieri

UE ..... Uniunea Europeană

EU ..... *European Union*

Capitolul 1:

Introducere

1.1 Context

Odata cu dezvoltarea sistemelor de calcul moderne si a componentelor Hardware, s-au putut realiza produse software complexe cu capacitati de stocare net superioare. Revolutia tehnologica a permis nu doar realizarea unor sarcini simple, precum calcule matematice, sau automatizarea unor dispozitive (de exemplu aprinderea automata a unui bec printr-un microcontroler), ci si posibilitatea gestionarii mai eficiente a informatiilor digitale (de la date bancare la fisiere media).

Aceasta a devenit treptat principala sursa legitima de inregistrare a oricarui tip de date (text, imagini, video, audio). Pentru a accesa si actualiza informatia digitala, s-au dezvoltat diferite versiuni de baze de date centralizate si distribuite.

Bazele de date centralizate sunt aplicatii software specializate ce folosesc resursele sistemului (a statiei) pentru a raspunde cat mai rapid interogarilor. Statiile trebuie sa detina multa putere de stocare si de procesare in comparatie cu un sistem de calcul normal destinat utilizatorilor casnici. In alta ordine de idei, s-au dezvoltat si baze de date distribuite, menite sa reduca din capacitatile tehnice ale serverului si sa stocheze informatia sub forma descentralizata. Cautarea resursei in acest context ar presupune interogarea recursiva a fiecarei entitati pana la gasirea sa. Prin acest mod, nu doar ca statiile pot avea si capabilitati tehnice mai reduse, dar si pot pastra copii de rezerva (backup) locale pentru fiecare segment de informatie in parte.

Această evoluție naturală spre descentralizare a deschis drumul unor concepte moderne precum învățarea automată federată (federated learning), unde datele nu mai sunt transferate către un server central. În schimb, modelele sunt antrenate local, iar parametrii sunt ulterior agregați global. Astfel, se menține confidențialitatea datelor, fără a compromite performanța modelului.

Evolutia tehnologica continua a dat nastere la o serie de atacuri cibernetice menite sa destabilizeze securitatea aplicatiilor si totodata sustragerea a cat mai multe date sau identitati private in contradictie cu normele legale. Cele mai populare atacuri raportate la scara globala pentru anul curent 2025 sunt ransomware (conform [1], in SUA s-au raportat cresteri de 149%), furtul de identitate prin exfiltrarea de credentiale, si phishing. Dezvoltarea modelelor de inteligență artificială a amplificat aceste riscuri, oferind atacatorilor instrumente automatizate pentru generarea și adaptarea atacurilor.

Pentru a limita utilizarea abuzivă a tehnologiilor bazate pe AI, Uniunea Europeană a adoptat în 2024 un set de reglementări stricte privind integrarea acestor module în aplicațiile software, prin AI Act [2].

Progresul din domeniul machine learning si a Large Language Models a fost posibil ca urmare a unui volum masiv de date disponibile si a nevoii tot mai mari de analiza. Acest lucru a determinat aparitia unei noi categorii de specialisti, data scientists, dedicati colectarii si prelucrarii minutioase a datelor pentru antrenarea modelelor.

Totuși, pe măsură ce investițiile în tehnologii AI au crescut, au apărut și actori rău intenționați care încearcă să exploateze vulnerabilitățile din procesul de antrenare. Întrucât modelele moderne depind de calitatea datelor folosite, acestea au devenit o țintă principală a atacurilor. Atacatorii se regasesc si ei intr-o pozitie constanta de adaptare la noile formalitati de securitate si incearca sa contracareze fiecare element nou. Astfel, avand in vedere complexitatea dezvoltarii unui modul de inteligenta artificiala specializat pe diferite domenii, tinta s-a redirectionat spre volumul de date pe care acestea le folosesc si care pot determina starea finala a aplicatiei.

În contextul învățării automate distribuite, literatura de specialitate identifică trei categorii majore de atacuri:

- Atacuri asupra datelor, precum data poisoning, unde setul de antrenare este manipulat pentru a altera comportamentul modelului;

- Atacuri asupra modelului, prin modificarea parametrilor sau a gradientului (de exemplu, model poisoning);

- Atacuri asupra canalului de comunicare, care vizează interceptarea sau modificarea mesajelor dintre entitățile participante.

Lucrarea de față se concentrează pe prima categorie, data poisoning, în cadrul unei infrastructuri de invatare federate.

1.2 Motivatia lucrarii

Avand in vedere aspectele legate de posibilitatea unei interventii asupra setului de date de antrenare, atac denumit otravire a datelor (data poisoning), munca cercetatorilor s-a ingreunat. Preocuparea nu mai este primordial asupra analizei setului de date de antrenare, cat despre mentinerea integritatii si a confidentialitatii lor. Pentru a răspunde acestor nevoi, colaborarea dintre cercetători s-a orientat către modele distribuite de lucru, iar învățarea federată (federated learning) a devenit una dintre principalele direcții. Aceasta permite colaborarea între participanți fără a partaja direct seturile lor de date, menținând o barieră naturală împotriva accesului neautorizat. Totuși, deși infrastructura este diferită față de abordările centralizate, vulnerabilitățile rămân, iar atacurile asupra datelor utilizate local pot afecta modelul global.

In urma unei analize proprii, am putut observa diferite solutii/frameworks de simulare a procesului de invatare automata federata, dar fara o integrare cu mecanisme moderne de testare pentru atacuri precum otravirea datelor (data

[1]https://www.dnsc.ro/vezi/document/buletin-de-indicatori-statistici-si-tendinte-de-securitate-cibernetica-h1-2025

[2]https://artificialintelligenceact.eu/wp-content/uploads/2024/11/Future-of-Life-InstituteAI-Act-overview-30-May-2024.pdf

poisoning) amintite anterior [3]. Unele dintre aceste framework-uri sunt poate dificil de gestionat si configurat [4], si nu permit extinderea usoara prin integrare altor componente. In acelasi timp, gandindu-ne la multitudinea de atacuri malware si la platformele de detectie a lor, devine clar ca in domeniul inteligentei artificiale lipseste o platforma centralizata, flexibila, dedicata testarii si evaluarii cu diferite tipuri de atacuri asupra modelelor distribuite.

Aceste limitări justifica realizarea prezentei lucrari, care își propune dezvoltarea unei platforme de simulare capabile sa testeze atacuri de tip data poisoning într-o infrastructura de învățare federată.

1.3 Obiectivele lucrarii

Plecand de la neajunsurile prezentate, ne propunem in aceasta lucrare sa venim in sprijinul comunitatii de cercetare stiintifica in domeniul securizarii solutiilor cu AI cu o platforma de simulare cu sursa deschisa ("open source"), a acestei clase de atacuri pe mai multe directii. Astfel, oferim cercetatorilor posibilitatea analizei algoritmului de antrenare propriu dezvoltat, plecand de la o retea neuronala de baza si un set de date uzual (imagini), si testarea sa prin antrenare in diferite conditii. Platforma in sine respecta toate normele unei aplicatii software de productie, in care fiecare actiune are propria sa logica de implementare. Serviciile sunt segregate suficient de mult incat sa permita o dezvoltare ulterioara prin integrarea lor cu alte sisteme.

Rezultatele pot fi utile in contextul securizarii procesului de antrenare al algoritmului, dar si pentru analiza factorilor de risc la care e expus in acest mediu.

Cercetatorul este cel care furnizeaza algoritmul python de antrenare a propriei retele neuronale sau algoritm de Machine Learning. El seteaza parametrii simularii atat pentru procesul de antrenare, cat si pentru tipul de atac de otravire a datelor. Platforma isi propune sa simuleze acest tip de atac cu ajutorul acestor setari de inceput intr-un mediu de invatare federata, furnizand la final o comparatie intre modelul antrenat folosind datele normale de antrenare si cel antrenat cu datele otravite. Aceste rezultate pot fi utile in semnalarea unui posibil risc la nivelul modelului dezvoltat, oferind mai apoi solutii de imbunatatire a implementarii sale.

1.4 Structura lucrarii

[3] https://ibmfl-api-docs.res.ibm.com/index.html
[4] https://github.com/IBM/federated-learning-lib/tree/main

Capitolul 2:

Notiuni Teoretice

In acest capitol, vor fi prezentate notiunile teoretice specifice intelegerii procesului de dezvoltare a platformei de simulare. Vom incepe cu Notiunile introductive despre conceptele de Machine Learning in antiteza cu Deep Learning. In continuare, vom discuta despre invatarea federata si arhitectura unei infrastructuri federate de invatare automata, tipurile de atacuri data poisoning implementate in procesul de simulare a atacurilor, precum si alte notiuni specifice implementarii.

2.1 Notiuni introductive

Machine Learning si Deep Learning sunt doua ramuri importante ale Inteligentei Artificiale care au rolul dezvoltarii unor modele specifice rezolvarii unor anumite actiuni. Pornind de la antrenarea de retele neuronale, ne orientam atentia spre setul de date de antrenare si spre actorii ce pot interveni in acest proces. Mediul in care testam ofera o perspectiva reala asupra impactului pe care il pot avea aceste atacuri la nivelul unei organizatii sau aplicatii.

2.1.1 Diferenta dintre Machine Learning si Deep Learning

Inteligența Artificială (AI) este domeniul vast care înglobează orice tehnică ce permite calculatoarelor să imite comportamentul uman. Informatia a evoluat treptat odata cu imbunatatirea capabilitatilor de stocare ale dispozitivelor si aparitia programelor software complexe. De la simplul format de text, inregistrari audio, pana la imagini si video in rezolutii 4K, modul de lucru s-a diversificat constant.

La fel au evoluat si cerintele utilizatorilor, care tind sa acceada catre solutii automate care sa le rezolve problemele uzuale, precum identificarea de patterns in imagini sau chiar din video, sau generarea de text.

IA vine sa rezolve aceste probleme si sa introduca algoritmi de rezolvare specifici pentru fiecare tip de informatie furnizata.

Machine Learning este o componenta importanta din domeniul IA care se diferentiaza de alte metode de antrenare prin optimizarile pe care le aduce erorilor ce apar din predictia rezultatului corect. Modelele de ML clasice se bazeaza pe interventia umana in factorul de decizie (supervised learning), mai precis datele de intrare sunt etichetate pentru a oferi un context de predictie stabil.

Deep Learning este o subcategorie a Machine Learning, care are rolul de a minimiza interventia umana si a automatiza procesul de decizie. Prin aceasta metoda se automatizeaza mare parte din extragerea caracteristicilor pe setul

de date, eliminand nevoia de a defini etichete pentru fiecare valoare de intrare (unsupervied learning).

Diferenta dintre aceste doua concepte este in modul in care acesti algoritmi invata si procentul de utilizare a datelor [1]. Scopul principal al invatarii automate este predictia. Pe baza unui set de date de antrenare si de testare, se determina o anumita categorie de iesire predefinita.

2.1.2 Retea Neuronala

Retelele Neuronale sunt un subset al Machine Learning si se identifica drept infrastructura de baza din cadrul algoritmilor de Deep Learning. Denumirea de "neuronal" se refera la structura lor interna, in care fiecare caracteristica (feature) este un neuron ce interactioneaza unii cu altii. Ele sunt compuse din 3 straturi/layers: primul strat il reprezinta stratul nodurilor de intrare, al doilea strat este denumit "stratul ascuns" (hidden layer) pt ca incapsuleaza mai multe straturi, iar ultimul strat este cel de iesire in care se face predictia propriu-zisa. Straturile ascunse sunt concepute pentru a procesa iterativ datele pornind de la starea lor din nodurile de intrare pana la stratul de iesire.

2.2 Invatare automata federata

Evolutia hardware in tehnologie a condus la cresterea numarului dispozitivelor mobile (telefoane, tablete), denumite gadgets din faptul ca sunt mici, portabile si moderne. Ele au fost mai departe adoptate la scara larga, devenind obiecte indispensabile in era tehnologica ce avea sa vina.

2.2.1 Concept

Invatarea automata federata permite lucrul cu modele de ML sau chiar retele neuronale, antrenate distribuit, pe un numar mare de dispozitive in scopul rezolvarii unei probleme de IA. Distribuirea sarcinilor a fost adoptata si in contextul opozitiei lucrului centralizat, pe servere ce detin capabilitati Hardware performante (placi grafice de ultima generatie), dar care genereaza costuri mari si care pot fi predispuse la amenintari de securitate cibernetica, fiind considerate SPOF(Single Point of Failure).

2.2.2 Arhitectura FL

In literatura, exista mai multe categorii de arhitecturi de invatare automata federata. In aceasta sectiune ne vom concentra pe clasificarea generala a arhitecturii unei aplicatii folosind federated learning, si vom enumera pentru o anumita categorie cum se clasifica dispozitivele utilizate.

Federated Learning, asa cum a mai fost mentionat, este organizat dintr-un server (agregator) si multiple dispozitive client. Modul in care aceste entitati comunica este fundamentul principal in modului de imbunatatire al invatarii.

In modul clasic al federated learning, dispozitivele client transmit actualizari ale modelului de baza la un server central care aplica asupra lor o functie de agregare, reconstruind intreg modelul de baza. Aceasta setare/model, presupune de fapt o delegare a sarcinii de invatare, insa pastreaza entitate centrala necesara imbunatatirii solutiei. Acest fapt, nu tine sa evita posibilitatea amenintarilor cibernetice (Single Point of Failure), ci doar sa usureze costurile centralizatorului in a procesa local problema, distribuind sarcinile.

Modelul Fully decentralized (peer-to-peer) learning, ofera o noua abordare si rezolva problema cibernetica amintita. In aceasta setare nu exista agregator, imbunatatirile fiind comunicate intre clienti interconectati. Ideea principala se bazeaza pe inlocuirea comunicarii cu agregatorul cu cea intre dispozitive individuale printr-un protocol prestabilit. In functie de numarul de dispozitive, se concepe un graf de conexiuni in care fiecare nod reprezinta un client, iar fiecare muchie un canal de comunicatie. Restrictia principala este ca un dispozitiv sa fie conectat la un numar maxim limitat de dispozitive adiacente, prestabilit, in contradictie cu un graf complet (stea) specific arhitecturii clasice client-server. Nodurile isi imbunatatesc propriile variante ale retelei, si isi comunica rezultatele pe care le agrega local, realizand o medie a ponderilor. In comparatie cu modelul federated learning clasic, modelul fully decentralized nu specifica de la inceput dispozitivelor un model de baza global de la care sa porneasca in procesul de rezolvare a problemei.

	Federated learning	**Fully decentralized (peer-to-peer) learning**
Orchestration	A central orchestration server or service organizes the training, but never sees raw data.	No centralized orchestration.
Wide-area communication	Typically a hub-and-spoke topology, with the hub representing a coordinating service provider (typically without data) and the spokes connecting to clients.	Peer-to-peer topology, with a possibly dynamic connectivity graph.

Figura 2.1: Imaginea 2.2.2: Modele Federated Learning

Imaginea de mai sus ofera o privire de ansamblu asupra celor doua modele de arhitecturi si caracteristicile acestora.

Model	Avantaje	Dezavantaje
Federated learning (centralized coordination)	• mai simplu de configurat (topologie hub-and-spoke) • pornește de la un model global de bază • agregarea centralizată reduce sarcina de calcul pe clienți • necesită mai puține conexiuni (doar client → server) • gestiunea și monitorizarea sunt mai simple	• SPOF (Single Point of Failure) – serverul central • serverul poate deveni țintă pentru atacuri • nu elimină riscurile cibernetice, doar distribuie munca • dependența de coordonator pentru progresul antrenării • necesită infrastructură centralizată permanent disponibilă
Fully decentralized / peer-to-peer learning	• previne atacurile specifice unui server central (evită SPOF) • reziliență crescută – compromiterea unui nod nu destabilizează întregul sistem • îmbunătățirile se propagă prin graf, fără entitate centrală • agregare locală (fiecare nod mediază ponderile) • poate scala natural dacă graful este bine proiectat	• necesită conexiuni suplimentare între clienți • topologie complexă, dificil de administrat • nu există model global inițial furnizat tuturor • performanța depinde de calitatea grafului de conexiuni • nodurile malițioase pot influența direct vecinii • necesită protocoale suplimentare pentru consistența actualizărilor

Tabel 2.1: Compararea modelelor Federated Learning și Fully Decentralized Learning

In tabelul de mai sus, sunt evidentiate avantajele si dezavantajele utilizarii celor doua tipuri de modele federated learning.

In continuarea acestei lucrari, se va discuta preponderent despre modelul clasic federated learning, fiind unul adoptat la scara larga si care ofera performante bune raportat la costurile de productie.

Modelul clasic la randul sau, se catalogheaza in literatura in functie de tipul dispozitivelor care iau parte la procesul de antrenare. Sub acest filtru, exista Cross-Device Federated Learning si Cross-Silo Federated Learning.

Cross-Device Federated Learning sunt dispozitivele client IOT uzuale, individuale, care comunica orchestratorului printr-un protocol prestabilit.

Cross-Silo Federated Learning sunt dispozitive din institutii guvernamentale, companii, sau centre de date distribuite geografic. Institutiile nu doresc sa schimbe informatii intre ele sau cu un furnizor de servicii central, pastrandu-si confidentialitatea, folosind federated learning pt a antrena propriul model pe datele private ale fiecaruia.

2.2.3 Procesul de antrenare FL

Primul pas este stabilirea conexiunii dintre dispozitive si un server de agregare ce permite antrenare distribuita a tipului de retea neuronala sau model ML specific problemei. Odata stabilit canalul, in faza de configurare initiala, serverul trimite dispozitivelor starea de baza a retelei neuronale, ponderile, in vederea antrenarii individuale. Fiecare retea se antreneaza cu datele extrase local (on device) si isi imbunatateste configuratia interna la fiecare epoca pentru o perioada de timp bine determinata.

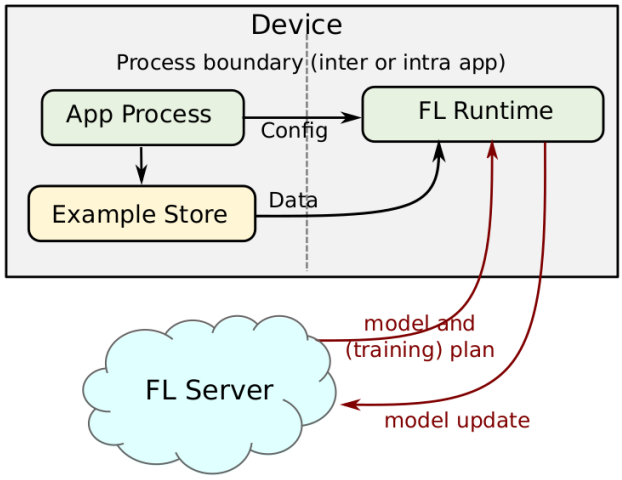

Figura 2.2: Imagine 2.2.3.1: Arhitectura interna a unui dispozitiv

Figura de mai sus descrie operatiile specifice programului Software care se ocupa de antrenarea retelei/modelului. Putem observa cum dispozitivele pri-

mesc un plan de antrenare de baza pe care il vor antrena local pe un set de date limitat.

Desi acest pas nu aduce un procent de imbunatatiri foarte mari, in faza urmatoare, dispozitivele vor transmite configuratiile curente ale retelelor lor la orchestrator (server). Rutina FL_Runtime extrage configuratia noua locala si ii comunica serverului pentru o posibila actualizare a sa.

In cele din urma, entitatea centrala combina toate aceste ponderi aplicand o functie de agregare si in cazul imbunatatirii setului de ponderi, modifica configuratia de baza si o retransmite dispozitivelor pereche. Daca ponderile noi nu se imbunatatesc semnificativ fata de configuratia de baza, atunci se patreaza aceasta din urma, iar in caz contrar se actualizeaza cu noile ponderi.

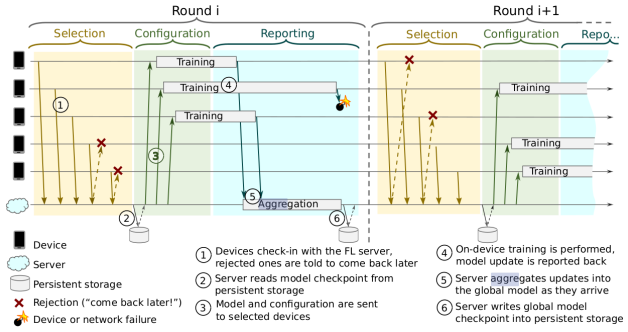

Figura 2.3: Imagine 2.2.3.2: Procedee in invatare automata federata

In figura de mai sus, se pot observa intr-o maniera continua, fluxul de comunicatie dintre dispozitive si serverul agregator, precum si operatiile specifice fiecarei entitati dintr-o runda de mesaje.

Securitatea protocoalelor de agregare, utilizate in comunicatii dintre clienti si orchestrator, este o componenta importanta in procesul federated learning. De mentionat este faptul ca, in aceasta topologie, comunicatiile au loc criptat, folosind metode specifice precum criptare homomorfica, sau chiar OTP, insa securitatea datelor de pe dispozitive ramane la latitudinea acestuia.

2.2.4 Exemple in viata reala

Federated Learning s-a extins rapid în numeroase domenii datorită capacității sale de a antrena modele performante fără a colecta sau centraliza date sensibile. Prin păstrarea informațiilor la nivelul fiecărui dispozitiv sau instituții, FL reduce riscurile asociate scurgerilor de date și permite colaborarea între entități care altfel nu ar putea împărtăși date brute. În continuare sunt prezentate câteva exemple reprezentative ale utilizării sale în aplicații din lumea reală.

Industrie și IoT

- **Mentenanță predictivă:** Vehiculele moderne, utilajele industriale și echipamentele IoT generează constant date despre starea componentelor. FL permite antrenarea unui model comun care poate prezice momentul oportun pentru realizarea mentenanței fără a colecta date brute de la fiecare dispozitiv.
- **Dispozitive de monitorizare:** Senzori purtabili și dispozitive smart home pot furniza statistici privind activitatea sau consumul energetic, păstrând datele utilizatorilor la sursă.

Medical

- **Diagnostic, prognoză și imagistică:** FL este folosit în spitale și clinici pentru detectarea celulelor canceroase din imagini RMN, CT sau radiografii, fără transferul imaginilor către un server central.
- **Confidențialitate menținută la sursă:** Fiecare instituție medicală antrenează local o parte din model, partajând doar actualizările, ceea ce permite colaborarea fără a încălca regulile privind datele pacienților.

Financiar

- **Detectarea fraudelor:** Instituțiile financiare pot îmbunătăți detectarea tranzacțiilor suspecte analizând tipare comune fără a expune date sensibile despre clienți.

Servicii și experiență utilizator

- **Recomandări personalizate:** Platformele de streaming și aplicațiile mobile generează recomandări local, pe dispozitiv, fără a trimite istoricul complet al utilizatorului către server.
- **Analiză comportamentală:** FL poate analiza activitatea utilizatorilor pentru a sugera rutine sănătoase sau îmbunătățiri ale stilului de viață, păstrând confidențialitatea datelor.

Securitate și privacy

- **Supraveghere fără expunerea datelor sensibile:** Modelele de recunoaștere facială pot fi antrenate fără a transmite imagini reale, doar parametrii aferenți.
- **Analiză a sentimentelor:** FL poate analiza reacțiile utilizatorilor la evenimente sociale (like-uri, share-uri, comentarii) fără colectarea directă a acestor date de către platformă.

2.3 Data Poisoning

2.4 Alte Notiuni

Capitolul 3:

Proiectare, Implementare si Testare

3.1 Cerintele Software

3.1.1 Cerintele functionale

3.1.2 Cerintele nefunctionale

3.2 Arhitectura platformei

3.2.1 Containere

3.2.2 Server

3.3 Testare

Capitolul 4:

Rezultate si Metrici Simulari

4.1 Evaluare Performante

4.1.1 Scalabilitatea Simularilor

4.1.2 Scalabilitatea platformei

4.2 Evaluare Rezultate

4.2.1 Performante Gaussian Noise

4.2.2 Performante Label-Flip

4.2.3 Performante Backdoor

Capitolul 5:

Concluzii si dezvoltare ulterioara

5.1 Starea Curenta

5.2 Dezvoltare Ulterioara

5.3 Tabele

Tabelele sunt aranjări a informației într-o structură formată din linii și coloane, care permite o mai bună observare a acesteia.

Mai jos apar două exemple. Primul tabel este de dimensiune mică. Al doilea, din cauza dimensiunii mai mari, are o orientare inversată și este plasat singur pe o pagină.

Nume Complet	Funcție Ocupată
Joshua Roob	Manager de Proiect
Asa Hauck	Artist Grafic
Harley Hagenes	Programator

Tabel 5.1: Colaboratori la Realizarea Studiului

Stat	Oraș	Latitudine	Longitudine
South Carolina	Corwinberg	86.609523	42.408007
Rhode Island	East Isaacmouth	63.17309	-13.786023
Mississippi	North Noblestad	-31.316834	5.280483
Illinois	Grahamland	-39.853659	-77.713676
Rhode Island	West Richardfurt	67.583131	31.858455
Florida	Port Roberta	25.276026	83.715344

Tabel 5.2: Locații de Conducere a Studiului

5.4 Imagini

Imaginile sunt utilizate în cadrul lucrării pentru exemplificarea unor idei în manieră vizuală.

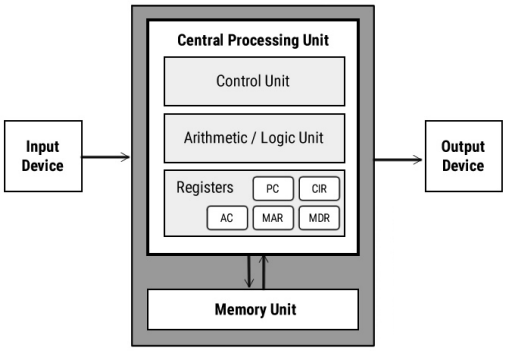

Figura 5.1: Arhitectura unui calculator[1]

5.5 Liste

Listele sunt simple serii de informații.

- Un item
- Unul dintre itemi
- Încă un item

Acestea pot conține itemi identificați prin numere dacă indexarea sau sortarea sunt necesare.

1. Primul item
2. Al doilea item
3. Al treilea item

5.6 Formule Matematice

LaTeX oferă un mod programatic de a construi formule matematice, după cum este cea de mai jos.

[1] Arhitectura ilustrată este de fapt cea von Neumann.

$$\sum \mathbf{F} = 0 \Leftrightarrow \frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t} = 0$$

5.7 Note de Subsol. Citări

Notele de subsol pot fi utile în cazul explicațiilor suplimentare (cum a fost cea referitoare la imaginea inclusă, la care sintaxa este puțin diferită din cauza plasării notei în cadrul legendei) sau a citărilor[2] care nu se pretează a fi trecute în bibliografie din cauza utilizării lor punctuale.

Pe de altă parte, sursele bibliografice citate intens [**cloud_crypto**] sunt marcate corespunzător și notate în bibliografie.

5.8 Etichete. Referințe

În cadrul surselor LaTeX a acestui document, apar *tag*-uri `\label` care creează o etichetă utilă referințelor interne. Acestea din urmă indică elemente din cadrul documentului curent (de exemplu, către tabelul 5.1).

Mai pot apărea referințe externe, către resurse din Internet (de exemplu, către *website*-ul Wikipedia).

[2] **morphological_operations**

Bibliografie

Cărți

Articole Științifice