UNIVERSITATEA DIN BUCUREȘTI

FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ

LUCRARE DE LICENȚĂ

Absolvent

Tănase Victor-Flavian

Coordonator științific

Prof. Dr. Radu Ionescu

București, iunie 2024

UNIVERSITATEA DIN BUCUREȘTI

FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ

DETECȚIA DE CONȚINUT DEEPFAKE CU AJUTORUL REȚELELOR NEURONALE ADÂNCI

Absolvent
Tănase Victor-Flavian

Coordonator științific
Prof. Dr. Radu Ionescu

București, iunie 2024

Rezumat

Evoluția capabilităților algoritmilor de inteligență artificială din ultimii ani a reinventat crearea de conținut în sfera digitală. În același timp, progresul în câmpul vederii artificiale a fost văzut de către unele entități ca o oportunitate de a răspândi dezinformare sau de a crea conținut malițios greu de detectat cu ochiul liber.

Ceea ce a început ca cercetare academică realizată de către Justus Thies et al. [11] a dat naștere unui mijloc de fabricare de conținut contrafăcut, capabil să manipuleze imagini și videoclipuri cu un realism nemaivăzut. Acest tip de conținut poate fi folosit in diverse scopuri cu rea-voință precum: șantaj, manipulare politică, înșelătorie, furt de identitate sau creare de conținut neadecvat, de cele mai multe ori aceste atacuri vizând persoane publice.

În scopul combaterii dezinformării, această lucrare are ca obiectiv punerea la dispoziție către public a unui model de inteligență artificială cu o interfață web, ce poate fi ușor de utilizat pentru detectarea sau cel puțin ridicarea suspiciunii asupra imaginilor sau videoclipurilor posibil fabricate.

Abstract

The evolution of the artificial intelligence models in recent years has reinvented the creation of content on social media. In the same time, the progress in Computer Vision has been viewed by others as an opportunity to spread misinformation and create malicious content that can be hardly detected by the naked eye.

What has started as academic research by Justus Thies et al. [11] pioneered a new way to fabricate content, capable to manipulate images and videos with unprecedented realism. This type of content can be used in many harmful ways such as blackmailing, political manipulation, fraudulent schemes, identity theft, and the creation of explicit material. Often, public figures are the primary targets of such attacks.

In order to fight disinformation, this work aims to provide the public with an artificial intelligence model with a web interface, that can be easily used to detect or at least raise suspicion over possibly fabricated images or videos.

Cuprins

Capitolul 1

Introducere

1.1 Descrierea problemei

1.2 Motivație

1.3 Contribuție personală

1.4 Privire de ansamblu și organizarea lucrării

Acest capitol abordeaza conceptele fundamentale care stau la baza înțelegerii rețelelor neuronale.

Capitolul 2

Concepte Teoretice

2.1 Rețele neuronale

Rețelele neuronale stau la baza tuturor algoritmilor moderni de inteligență artificială. Acestea au revoluționat industria învățării automate prin puterea lor de a simula funcțiile cognitive ale creierului uman, fiind des folosite in invățarea de tipare(în engleză pattern recognition), diferite sarcini de clasificare si predicție. Ele pot fi văzute ca funcții complicate care primesc date de intrare sub forma unor valori numerice și returnează valori reprezentative pentru acestea.

2.1.1 O scurtă istorie a rețelelor neuronale

- În lucrarea intitulată „A Logical Calculus of the Ideas Immanent in Nervous Activity"(1943) [6] Warren McCulloch și Walter Pitts au fost primii care au teoretizat o implementare matematică simplificată a neuronului ca o poartă logică binară. Aceștia au demonstrat teoretic faptul că neuronii au capacitatea de a reprezenta orice funcție matematică.

- În 1958, este publicată lucrarea cercetătorului Frank Rosenblatt „The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain" [9], în care acesta introduce conceptul de Perceptron, cel mai simplu model de rețea neuronală.

- Spre finalul anilor 1980, după o lungă perioadă în care studiul rețelelor neuronale a fost neglijat, David E. Rumelhart, Geoffrey Hinton, and Ronald J. Williams (1986) [10] prezintă **backpropagarea**, un algoritm prin care o rețea neuronală putea să învețe eficient, adresând multe dintre problemele ridicate până la acel moment.

- La începutul anilor 2000, tot Geoffrey Hinton, de data aceasta alături de Ruslan Salakhutdinov [5] aduc o soluție pentru problema gradienților care dispar, prin reducerea dimensionalității datelor.

Aceste lucrări din literatură au ajutat la transformarea unor concepte pur teoretice în unelte cu putere de învățare și de decizie care au ajutat la modelarea industriei moderne și au împins progresul tehnologic către noi limite.

2.1.2 De la neuronul biologic la cel artificial

La fel ca multe dintre invențiile revoluționare de-alungul istoriei, cercetătorii au avut ca sursă de insipirație viețuitoarele. În cazul inteligenței artificiale, aspirația cercetătorilor a fost să replice neuronul biologic.

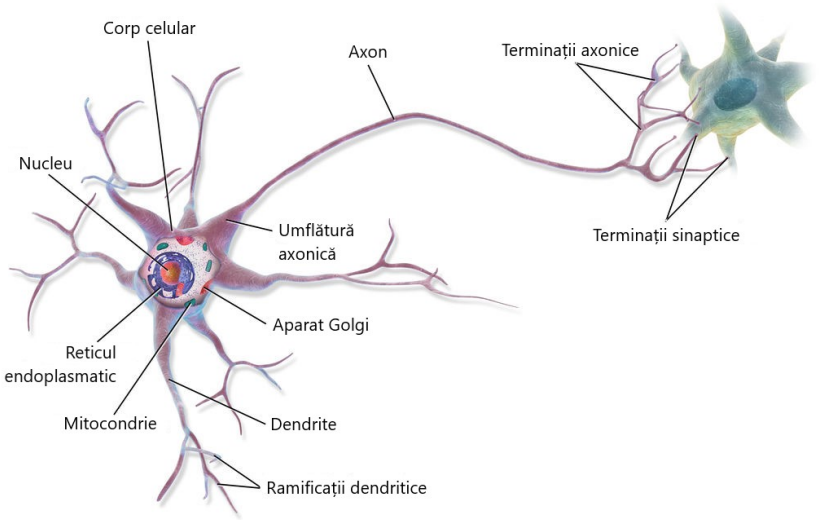

Figura 2.1: Anatomia unui neuron [1]

În zona centrală a neuronului se află corpul celular(soma) care conține nucleul, locul care înmagazineaza tot materialul genetic al celulei. Legate de corpul celular

sunt dendritele, care interceptează semnale chimice de la alți neuroni. Atașat de corpul celular, se află o prelungire alungită numită axon, al cărei scop este să transmită semnale electrice către alți neuroni sau către țesuturi. La capătul axonului se găsesc terminațiile acestuia, care sunt legate de dendritele sau de corpul celular ale altui neuron.

În creierul biologic, se găsesc miliarde de neuroni, fiecare având mii de legaturi cu alți neuroni aceștia fiind dipusi in straturi pentru a crea țestul nervos. Cu toate că neuronul in sine, nu funcționează intr-un mod complex, ansamblul a miliarde de mecanisme simple într-o rețea imensa are capabilități impresionante.

Plecând de la aceste premise Warren McCulloch și Walter Pitts(1943) [6] au prezentat în lucrarea lor revoluționară o versiune simplificată a neuronului biologic. Aceștia au văzut neuronul artificial ca o pe poartă logică, cu mai multe intrări și o singură ieșire. Un semnal electric era transmis mai departe doar dacă neuronul avea un număr de intrări care trecea de un anumit prag, bazându-se pe principiul de totul sau nimic al neuronilor biologici.

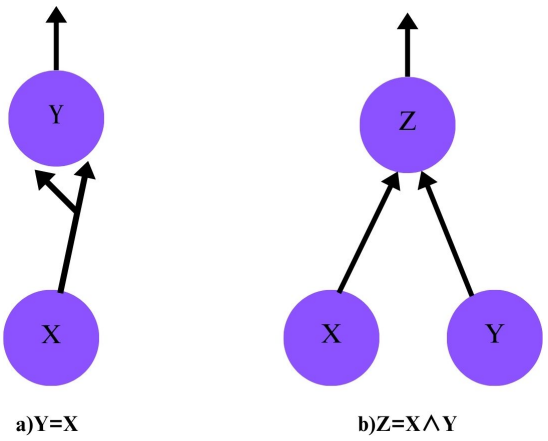

Figura 2.2: Neuroni artificiali calculând operatii logice

a) Neuronul Y nu poate transmite semnalul mai departe doar cu un singur impuls

b) Neuronul Z are nevoie de ambii neuroni X și Y pentru a trimite un semnal mai departe

2.1.3 Perceptronul

În 1958 Frank Rosenblatt [9] introduce in literatură **perceptronul**, cel mai simplu model de rețea neuronală. Acesta duce neuronul artificial propus de către McCulloch și Pitts [6] la un alt nivel. Comparat cu neuronul care transmitea doar semnale binare, cel propus de Rosenblatt este capabil sa proceseze și să transmită mai departe valori numerice. Scopul final este de a organiza aceste unități de calcul într-o „rețea" care primește datele de intrare sub formă numerică și returnează o valoare care le reprezintă.

Într-o astfel de rețea fiecare neuron se folosește datele de intrare pentru a produce o valoare de ieșire. Fiecărei valori de intrare îi corespunde o pondere(în engleză: weight), un număr real al cărui scop este sa controleze contribuția sa la informația transmisă mai departe de către neuron. Aceste valori sunt însumate ponderat, iar la valoarea obținută se adaugă un neuron special, care la rândul lui are o pondere. Acest neuron se numește *bias*, un număr real, notat cu b, care elimină constângerea ca funcția ce reprezintă datele să treacă prin originea planului sau a hiperplanului. Rezultatul reprezintă o funcție care este dată ca parametru unei funcții liniare numită „funcție de pas"(in engleză: step function) pentru a produce valoarea de ieșire a perceptronului.

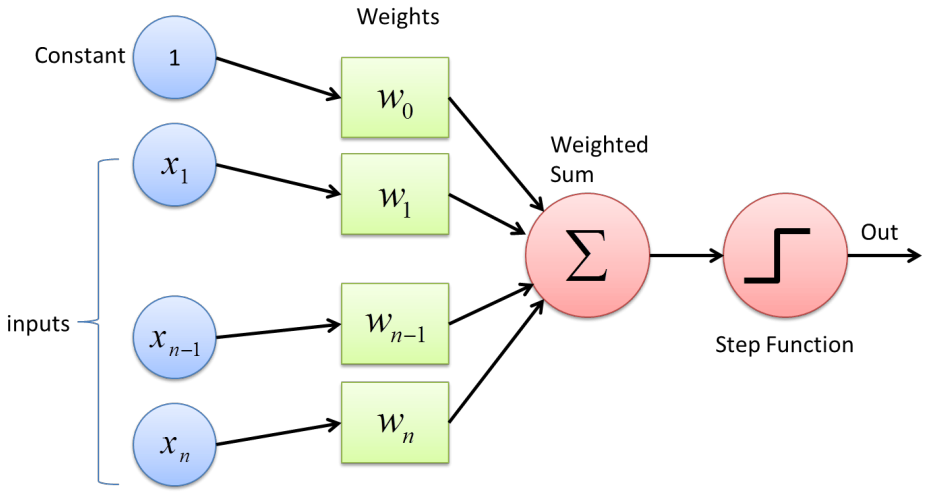

Figura 2.3: Perceptronul [8]

Notații:

- $x = (x_1, x_2, x_3..., x_n)$ datele de intrare
- $w = (w_1, w_2, w_3, ..., w_n)$ ponderile corespunzătoare
- $b =$ totalitatea neuronilor de *bias*
- $z =$ combinația liniară dintre x si w
- $z = w_1x_1 + w_2x_2 + ... + w_nx_n = \sum_{i=1}^{n} w_ix_i + b = w^Tx + b$
- $step(z) =$ *output-ul* perceptronului

O funcție comună folosită drept *step function* este funcția semn definită astfel:

$$\text{sgn}(x) = \begin{cases} -1 & \text{if } x < 0, \\ 0 & \text{if } x = 0, \\ 1 & \text{if } x > 0. \end{cases}$$

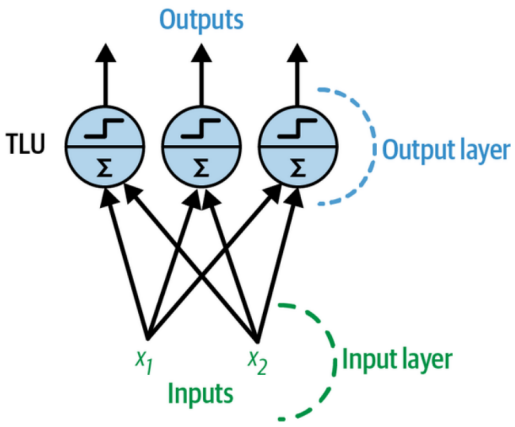

Figura 2.4: Un perceptron capabil să separe date în mai multe clase [3]

În Figura 2.4 este prezentată o arhitectură de perceptron capabilă sa distingă între 3 clase. Datele de intrare, in cazul nostru $x1$, $x2$ reprezintă *stratul de intrare*. Fiecare valoare din stratul de intrare este legată de toți neuronii din stratul urmator, fiecare legatură având o pondere individuală. Stratul care produce valorile de ieșire se numește *stratul de ieșire* și in exemplul prezentat este format din 3 neuroni, corespunzători fiecărei clasificări posibile.

În ciuda optimismului declanșat de capabilitățile remarcabile la acea vreme a perceptronului, în 1969 Marvin Minsky și Seymour Papert [7] au demonstrat limitarea acestui model de a rezolva cateva probleme triviale, precum operația XOR.

2.1.4 Multilayer Perceptron(MLP)

Soluția pentru limitările perceptronului a fost unificarea mai multor rețele de tip perceptron intr-o rețea mai mare, numita Multilayer Perceptron(MLP). Noua clasificare a straturilor este :

- Stratul de input = stratul ce conține doar valorile de intrare
- Straturile ascunse = straturile prin care se propaga informația. Se sitează între stratul de input și cel de output
- Stratul de output = stratul care ne oferă rezultatul trecerii informației prin rețea

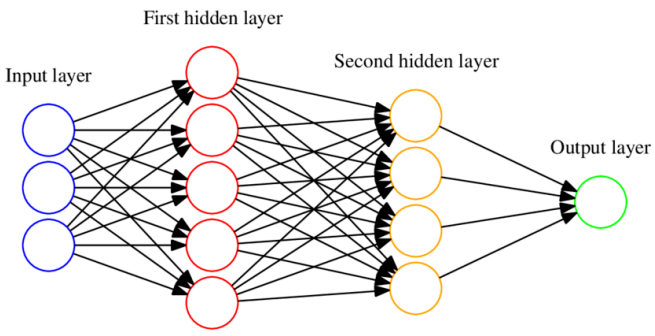

Figura 2.5: Arhitectura de tip MLP(Multilayer Perceptron) [2]

Arhitectura MLP este una de tip **feedforward**, cu alte cuvinte, informația se propagă prin rețea intr-o singură direcție, de la stânga la dreapta(dinspre input către output).

2.1.5 Backpropagarea

Algoritmul prin care o rețea neuronală învață se numește **backpropagation**. Acesta presupune trecerea prin rețea în repetate rânduri, în doi pași: unul inainte și unul înapoi. Dupa fiecare trecere parametrii(ponderile) sunt actualizați pentru a obține predicții mai bune. Algoritmul poate fi descris astfel:

- Setul de date este spart în blocuri mai mici, cele mai întâlnite dimensiuni fiind de 32, 64 sau 128 de instanțe per bloc
- Fiecare bloc este trecut prin rețea: se calculează output-ul primului strat, care este dat ca intrare pentru urmatorul strat, continuând în același fel până se ajunge la stratul de output. Acesta este pasul forward(*forward pass*).
- Se evaluează performanța predicțiilor cu ajutorul unei funcții numită **funcție de pierdere**. Scopul acestei funcții este să furnizeze un scor care să descrie cât de precise au fost predicțiile rețelei față de rezultatul dorit.
- Dupa calculul erorii, algoritmul calculează contribuția fiecărei ponderi din rețea la eroarea totală, cu ajutorul regulii fundamentale din analiza matematică: regula înlănțuirii(în engleză: *chain rule*).
- Contribuția la eroarea totală a unui parametru se numește **gradient**.Valoarea gradientului practic măsoară cât de mult influențează eroarea totală modificarea acelui parmetru.
- În cele din urmă, algoritmul folosește gradienții calculați pentru optimizarea parametrilor cu ajutorul unui algoritm de optimizare.

2.1.6 Algoritmul coborârii pe gradient(*Gradient Descent*)

Coborârea pe gradient este un algoritm iterativ de optimizare, folosit des în literatură, al cărui scop este de găsi minimul global al unei funcții. În cadrul rețelelor neuronale acesta este folosit pentru a optimiza parametrii rețelei astfel încat valoarea erorii totale să se apropie la fiecare iterație de minimul funcției de pierdere. Dupa calcularea gradienților, fiecare pondere se actualizeaza după formula:

$$w_{new} = w - \eta \frac{\delta L}{\delta w}$$

Unde:

- w_{new} = noua valoare a ponderii
- $\frac{\delta L}{\delta w}$ = gradientul
- η = rata de învățare(hiperparametru ce controleaza cât de mult se modifică valorile ponderilor)

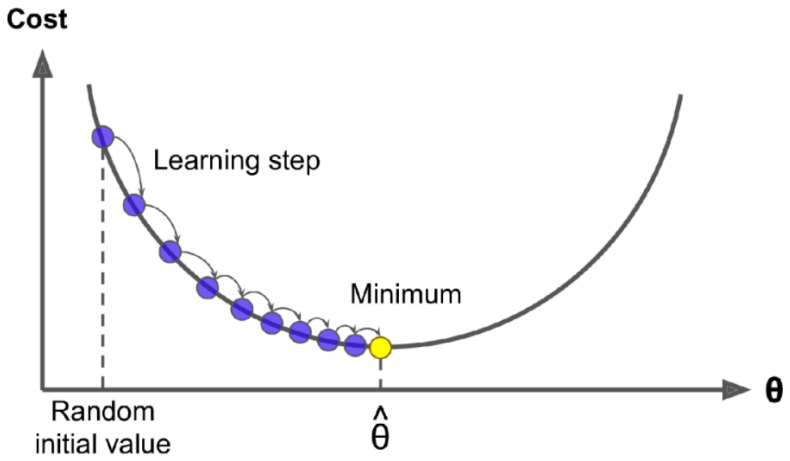

Figura 2.6: Vizualizare a algoritmului de coborârii pe gradient [4]

Bibliografie

[1] Bruce Blaus, *Multipolar neuron*, https://en.wikipedia.org/wiki/Neuron, Accesat: 7 Mai 2024, 2013.

[2] Matěj Fanta, „Rules extraction from deep neural networks Master thesis", Teză de doct., Aug. 2019, DOI: 10.13140/RG.2.2.22319.28324.

[3] Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*, ed. de Nicole Taché și Michele Cronin, O'Reilly Media, 2022.

[4] *Gradient Descent*, https://pantelis.github.io/cs634/docs/common/lectures/optimization/sgd/, Accesat: 9 Mai 2024.

[5] Geoffrey E Hinton și Ruslan R Salakhutdinov, „Reducing the dimensionality of data with neural networks", în *science* 313.5786 (2006), pp. 504–507.

[6] Warren S McCulloch și Walter Pitts, „A logical calculus of the ideas immanent in nervous activity", în *The bulletin of mathematical biophysics* 5 (1943), pp. 115–133.

[7] Marvin Minsky și Seymour Papert, „An introduction to computational geometry", în *Cambridge tiass., HIT* 479.480 (1969), p. 104.

[8] *Perceptron Definition*, https://images.deepai.org/glossary-terms/perceptron-6168423.jpg, Accesat: 8 Mai 2024.

[9] Frank Rosenblatt, „The perceptron: a probabilistic model for information storage and organization in the brain.", în *Psychological review* 65.6 (1958), p. 386.

[10] David E Rumelhart, Geoffrey E Hinton și Ronald J Williams, „Learning representations by back-propagating errors", în *nature* 323.6088 (1986), pp. 533–536.

[11] Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt și Matthias Nießner, „Face2face: Real-time face capture and reenactment of rgb videos", în *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2387–2395.