PhD Thesis / Teză de doctorat
Advanced Machine Learning Systems
Sisteme Avansate de Învăţare Automată
Semiotics and Information Theory in Neural Network Optimization
Semiotica şi Teoria Informaţiei în Optimizarea Reţelelor Neuronale

Rezumat

Student Doctorand: Bogdan-Adrian Muşat
Coordonator Ştiinţific: Prof. dr. mat. Răzvan Andonie

Capitolul 1

Introducere

1.1 Motivația și Importanța Cercetării

Teza de doctorat prezentată în această cercetare cuprinde trei direcții de cercetare semnificative: Semiotica, Teoria informației și Pruning-ul rețelelor neuronale, cu scopul de a avansa în domeniul învățării profunde și de a-i debloca potențialul în diverse domenii. Aceste trei direcții sunt interconectate și se pot completa reciproc pentru a îmbunătăți interpretabilitatea, eficiența și înțelegerea semantică a modelelor de învățare profundă în contextul datelor vizuale.

Una dintre problemele principale pe care ne concentrăm se referă la încorporarea semioticii în rețelele neuronale convoluționale (CNN). Acest efort este foarte promițător pentru a răspunde cerinței urgente de îmbunătățire a interpretabilității și explicabilității în domeniul procesării datelor vizuale [15, 123]. În timp ce CNN-urile excelează în analiza și extragerea caracteristicilor din datele vizuale, adesea le lipsește o înțelegere cuprinzătoare a contextului semantic subiacent încorporat în date [7, 29, 128, 129]. Acesta este punctul în care semiotica, un domeniu dedicat studiului simbolurilor și semnelor și al semnificației acestora, devine crucială. Prin încorporarea principiilor semiotice în arhitecturile CNN, cercetătorii au potențialul de a reduce acest decalaj și de a permite niveluri mai profunde de înțelegere.

Integrarea principiilor semiotice în arhitecturile CNN deschide posibilități interesante pentru îmbunătățirea înțelegerii semantice a modelelor CNN [23]. Luând în considerare semnificația și interpretarea simbolurilor și a semnelor în contextul datelor vizuale, aceste modele pot oferi interpretări și explicații mai semnificative pentru predicțiile și deciziile lor. Această integrare nu numai că oferă informații valoroase despre procesul de raționament al CNN-urilor, dar permite utilizatorilor și părților interesate să înțeleagă factorii care stau la baza rezultatelor modelului.

Cu toate acestea, se ridică o întrebare critică: Cum putem încorpora în mod eficient principiile semiotice în arhitecturile CNN? Deși beneficiile potențiale sunt clare, detaliile

de implementare și tehnicile de reprezentare și interpretare a simbolurilor și semnelor în cadrul straturilor și operațiunilor unui CNN rămân domenii deschise pentru explorare. Eforturile de cercetare s-ar putea concentra pe dezvoltarea unor metodologii care să integreze fără probleme principiile semiotice în cadrele CNN existente, permițând o fuziune armonioasă a procesării datelor vizuale și a interpretării semnificative [116].

Abordând această întrebare, cercetătorii ar putea deschide calea pentru modele CNN care nu numai că excelează în capacitatea lor de a extrage caracteristici vizuale, dar posedă și o înțelegere mai profundă a contextului semiotic în care există aceste caracteristici. Acest lucru ar conduce la rezultate mai fiabile și mai explicabile, consolidând încrederea și promovând o adoptare mai largă a modelelor CNN în diverse domenii în care interpretabilitatea este de o importanță majoră. Obiectivul nostru în această teză este de a aborda provocarea de a încorpora în mod eficient conceptele semiotice în CNN-uri.

O altă provocare pe care ne străduim să o abordăm implică integrarea cadrului Information Bottleneck (IB) cu semiotica, oferind potențialul unor avantaje suplimentare în ceea ce privește interpretabilitatea și eficiența. Cadrul IB, care își are rădăcinile în teoria informației, se concentrează pe extragerea informațiilor relevante și semnificative, eliminând în același timp detaliile irelevante [93, 110, 111]. Prin combinarea cadrului IB cu semiotica, cercetătorii pot valorifica principiile extragerii informațiilor semnificative pentru a ghida interpretarea și înțelegerea simbolurilor și semnelor codificate în date.

Această integrare joacă un rol crucial în ghidarea procesului de îmbogățire a informațiilor în cadrul modelelor de învățare profundă. Ea ajută la identificarea celor mai relevante semne și simboluri care contribuie cel mai mult la rezultatul dorit. Prin încorporarea principiilor semiotice, interpretarea acestor semne și simboluri devine mai semnificativă și mai interpretabilă, permițând o înțelegere mai profundă a datelor subiacente.

Cu toate acestea, se ridică o întrebare deschisă: Cum putem găsi un echilibru în timpul procesului de îmbuteliere a informației, asigurând o compresie eficientă, păstrând în același timp integritatea semnificației semiotice? Este crucial să se evite compromiterea înțelegerii semantice esențiale încorporate în semne și simboluri. Eforturile de cercetare ar trebui să exploreze tehnici de optimizare care să ia în considerare atât principiile teoretice ale informației, cât și principiile semiotice. Încă o dată, obiectivul nostru este de a aborda această întrebare nerezolvată și de a determina dacă cercetarea noastră o poate aborda în mod eficient.

În cele din urmă, ultima provocare pe care ne propunem să o abordăm se referă la integrarea tunderea rețelelor neuronale și a teoriei informației. Tehnicile de tăiere a rețelelor neuronale oferă o soluție pentru a optimiza eficiența modelului și complexitatea computațională prin eliminarea selectivă a componentelor inutile sau redundante din rețelele neuronale profunde [40, 46, 56, 61]. Aceste rețele au adesea un număr mare de parametri, ceea ce le face costisitoare din punct de vedere computațional și consuma-

toare de resurse. Integrarea teoriei informației cu pruningul rețelelor neuronale ar putea aborda provocările legate de optimizarea modelului, complexitatea computațională și eficiența în învățarea profundă.

Teoria informației oferă perspective fundamentale în ceea ce privește reprezentarea, compresia și transmiterea datelor. Prin valorificarea teoriei informației, cercetătorii urmăresc să optimizeze fluxul de informații în cadrul rețelei și să îmbunătățească eficiența acesteia. Concepte precum entropia, informația reciprocă și capacitatea canalului sunt instrumente valoroase pentru a cuantifica conținutul de informații și redundanța din cadrul rețelei [18].

Combinarea teoriei informației și a reducerii rețelei neuronale permite identificarea și păstrarea celor mai informative componente, reducând în același timp complexitatea de calcul și amprenta de memorie. Măsurile teoretice ale informației pot evalua importanța diferitelor componente ale rețelei, permițând procesului de tăiere să rețină cele mai relevante caracteristici [16, 59, 68]. Această abordare ajută la abordarea unor provocări precum supraadaptarea și echilibrează complexitatea modelului cu eficiența datelor [70, 118].

Algoritmii eficienți de curățare care utilizează concepte teoretice ale informației pot accelera procesul. Prin utilizarea teoriei informației, cercetătorii pot identifica componentele mai puțin informative, permițând o tăiere mai rapidă și mai bine direcționată. Acest lucru reduce cheltuielile de calcul asociate cu formarea și reglarea fină a rețelelor curățate, sporind și mai mult eficiența în învățarea profundă. Care sunt modalitățile prin care algoritmii de tăiere eficientă pot valorifica conceptele teoriei informației pentru a accelera procesul de tăiere? În plus, cum pot cercetătorii să utilizeze în mod eficient tehnici specifice, bazate pe teoria informației, pentru a identifica componentele mai puțin informative? Aceste întrebări constituie o mare parte din cercetarea tezei noastre, deoarece ne propunem să descoperim noi strategii și abordări care pot spori eficiența și eficacitatea procesului de curățare în modelele de învățare profundă.

Integrarea semioticii, a teoriei informației și a pruningului rețelelor neuronale în această teză de doctorat oferă perspective valoroase și aplicații practice pentru domeniul învățării profunde. Prin examinarea conexiunilor dintre aceste direcții de cercetare, acest studiu oferă un cadru pentru a îmbunătăți interpretabilitatea, eficiența și înțelegerea semantică a modelelor de învățare profundă atunci când se analizează date vizuale. Aceste constatări contribuie la progresul continuu al domeniului și sunt promițătoare pentru investigații viitoare. Prin intermediul acestei cercetări, dobândim o înțelegere mai profundă a potențialului și a limitărilor învățării profunde, ceea ce ne aduce mai aproape de aplicații mai eficiente și mai semnificative în diverse domenii.

1.2 Rețele Neuronale Convoluționale

Rețelele neuronale convoluționale sunt un tip de rețea neuronală artificială care s-au dovedit a fi deosebit de eficiente în sarcini de viziune pe calculator, cum ar fi clasificarea imaginilor [42, 53, 97], detectarea obiectelor [60, 83] și segmentarea [14, 63, 84]. Acestea sunt inspirate de structura și funcția cortexului vizual din creier și utilizează o serie de straturi convoluționale pentru a extrage și a învăța caracteristici din imaginile de intrare.

Elementul de bază al unui CNN este stratul convoluțional, care aplică un set de filtre la imaginea de intrare și produce un set de hărți de caracteristici de ieșire. Aceste filtre sunt învățate în timpul procesului de instruire și sunt optimizate pentru a capta modele și structuri utile în datele de intrare. Hărțile de caracteristici de ieșire sunt apoi trecute prin straturi suplimentare, cum ar fi straturi de grupare, normalizare și straturi complet conectate, pentru a genera o predicție finală de ieșire.

CNN-urile au devenit un instrument utilizat pe scară largă și extrem de eficient în domeniul vederii computerizate, atingând performanțe de ultimă generație într-o serie de sarcini. CNN-urile au fost aplicate într-o varietate de aplicații, inclusiv conducerea autonomă [11,27], imagistica medicală [92] și recunoașterea facială [106]. În plus, CNN-urile au fost adaptate la alte domenii, cum ar fi prelucrarea limbajului natural [113] și analiza audio [47]. Versatilitatea și succesul CNN-urilor fac din ele un instrument valoros în multe domenii diferite de cercetare și în industrie.

În general, rețelele neuronale convoluționale au avut un impact semnificativ asupra domeniului vederii computerizate și reprezintă un instrument puternic pentru rezolvarea unei game largi de sarcini. Pe măsură ce cercetările în acest domeniu continuă, este probabil că vom asista la și mai multe progrese în ceea ce privește arhitecturile CNN și tehnicile de îmbunătățire a eficienței și performanței acestora.

1.3 Învățarea Prin Întărire

Învățarea prin întărire (RL) este un subdomeniu al învățării automate care se concentrează pe interacțiunea dintre un agent și mediul său. Aceasta implică învățarea modului de a lua decizii secvențiale pentru a maximiza un obiectiv pe termen lung. Această paradigmă de învățare a câștigat o atenție semnificativă în ultimii ani datorită capacității sale de a aborda probleme complexe cu un feedback rar, ceea ce o face potrivită pentru o gamă largă de aplicații, cum ar fi robotica, jocurile și sistemele autonome [102].

Q-learning este un algoritm popular utilizat în învățarea prin întărire, în special în scenarii cu spații de stare mari sau continue. Este un algoritm fără model care estimează direct valorile de acțiune, cunoscute și sub numele de valori Q. Valoarea Q a acțiunii a

în starea s, notată ca $Q(s, a)$, reprezintă recompensa cumulativă așteptată pornind de la starea s, luând acțiunea a și urmând o anumită politică π [102].

Combinația dintre învățarea prin întărire și rețelele neuronale profunde, cunoscută sub numele de învățare prin întărire profundă, a produs progrese remarcabile în domeniu. Deosebit de remarcabilă este atingerea unor performanțe de nivel uman de către agenți în sarcini complexe, cum ar fi jucarea jocurilor Atari și înfrângerea campionilor mondiali la jocuri de societate precum Go [67, 94]. Aceste realizări demonstrează în mod viu puterea învățării prin întărire în abordarea unor probleme dificile din lumea reală, deschizând, în consecință, noi posibilități pentru sistemele autonome și agenții inteligenți.

O altă paradigmă puternică care a apărut în RL este abordarea actor-critic. Algoritmii actor-critic combină elemente atât ale metodelor bazate pe valori, cum ar fi învățarea Q, cât și ale metodelor bazate pe politici. Aceștia mențin două componente separate: un actor care învață o politică și un critic care estimează funcția de valoare.

O variantă populară a algoritmilor actor-critic este metoda Advantage Actor-Critic (A2C). A2C combină avantajele atât ale metodelor bazate pe politici, cât și ale aproximării funcției de valoare. Aceasta utilizează mai mulți agenți actor-critic paraleli care interacționează cu mediul pentru a colecta date, care sunt apoi utilizate atât pentru actualizarea politicii, cât și pentru estimarea funcției valorii. Această paralelizare îmbunătățește eficiența eșantionării și permite actualizări mai stabile [66].

Algoritmul Deep Deterministic Policy Gradient (DDPG) este o metodă de învățare de întărire fără model, fără politică, care combină elemente atât ale algoritmului de învățare Q profundă, cât și ale algoritmului actor-critic [58]. DDPG este conceput special pentru a gestiona spații de acțiune continue, ceea ce îl face bine adaptat pentru sarcini cu control continuu [58, 95].

Algoritmul DDPG a găsit, de asemenea, aplicații dincolo de sarcinile tradiționale de învățare prin întărire. În contextul tăierii rețelelor neuronale, DDPG a fost integrat în cadrul AutoML for Model Compression (AMC) [45] ca metodă eficientă de ghidare a procesului de tăiere. AMC valorifică DDPG pentru a învăța o politică de selectare și tăiere a canalelor rețelei pe baza scorurilor de importanță ale acestora. Rețeaua de actori din DDPG este antrenată pentru a genera măști de tăiere care determină conectivitatea rețelei, în timp ce rețeaua critică estimează performanța modelelor tăiate. Prin intermediul procesului iterativ de instruire, DDPG explorează compromisul dintre dimensiunea modelului și performanță, descoperind în cele din urmă arhitecturi de rețea compacte și eficiente. Prin încorporarea DDPG în cadrul AMC, tunderea rețelelor neuronale devine o problemă de învățare prin întărire, permițând decizii de tundere automate și bazate pe date care au ca rezultat modele foarte comprimate fără pierderi semnificative de precizie [45]. Combinația dintre DDPG și cadrul AMC demonstrează versatilitatea tehnicilor de învățare prin întărire în abordarea problemelor complexe de optimizare în domeniul comprimării rețelelor neuronale și al eficienței modelelor.

Pe măsură ce cercetările în curs de desfășurare continuă să împingă limitele învățării prin întărire, aceasta deține un potențial extraordinar pentru abordarea problemelor complexe de luare a deciziilor în diverse domenii. De la vehicule autonome la sisteme de recomandări personalizate, învățarea prin întărire oferă un cadru puternic pentru crearea de agenți inteligenți care se pot adapta, învăța și excela în medii diverse. Viitorul învățării prin întărire pare promițător, având potențialul de a modela modul în care interacționăm cu tehnologia și de a deschide calea către o nouă eră a aplicațiilor bazate pe inteligență artificială.

1.4 Semiotica: Studiul Semnelor și Simbolurilor

Semiotica, cunoscută și sub numele de semiologie, este un domeniu de studiu care explorează modul în care semnele și simbolurile sunt folosite pentru comunicare. Acesta se concentrează pe analiza și interpretarea semnificațiilor acestor semne în diferite contexte. Semiotica recunoaște că semnele merg dincolo de cuvinte și imagini și includ alte experiențe senzoriale, cum ar fi sunetele, gesturile, mirosurile, gusturile și atingerea. Prin studierea modului în care semnele, contextele lor și interpretările oamenilor interacționează, semiotica ne ajută să înțelegem cum sunt create, negociate și împărtășite semnificațiile în cadrul diferitelor culturi și societăți.

În centrul semioticii se află conceptul de semn, care constă într-o formă fizică (semnificant) și conceptul mental pe care îl reprezintă (semnificat). Relația dintre semnificant și semnificat este arbitrară și se bazează pe convenții culturale. Aceasta înseamnă că nu există o legătură inerentă între ele; este un acord comun în cadrul unei anumite culturi sau comunități. De exemplu, în limba engleză, cuvântul "dog" reprezintă ideea unui animal cu patru picioare, în timp ce o altă limbă poate folosi un cuvânt diferit pentru același concept.

Semiotica se bazează pe lucrările lingvistului elvețian Ferdinand de Saussure [88]. Saussure a introdus ideea de semnificant și semnificat, subliniind importanța naturii relaționale a semnelor. El a susținut că semnificația provine din diferențele și relațiile dintre semne în cadrul unui sistem, mai degrabă decât din existența lor izolată. Acest concept a condus la noțiunea de sisteme de semne sau sisteme semnificante, în care semnele sunt organizate și structurate pentru a crea semnificație. Aceste sisteme de semne pot fi observate în diverse domenii, cum ar fi limbajul, artele vizuale, muzica și ritualurile sociale.

În plus față de abordarea structuralistă a lui Saussure, semiotica cuprinde diverse cadre și perspective teoretice. O figură influentă în semiotică este filosoful american Charles Sanders Peirce, care a propus un model triadic al semnelor [79]. Potrivit lui Peirce, semnele pot fi clasificate în trei tipuri: icoane, indici și simboluri. Icoanele sunt semne care seamănă sau imită obiectele pe care le reprezintă, cum ar fi o fotografie. Semnele index au o legătură cauzală sau contingentă cu referenții lor, cum ar fi fumul care indică un incendiu. Simbolurile, pe de altă parte, se bazează pe asociații arbitrare cu

semnificațiile lor, bazate pe convenții culturale comune. De exemplu, forma octogonală roșie a unui semn de oprire simbolizează oprirea în multe culturi.

Semiotica oferă un set de instrumente valoroase pentru analizarea și interpretarea diverselor forme de comunicare și a fenomenelor culturale [6]. Ea permite cercetătorilor să investigheze modul în care semnificația este construită, transmisă și înțeleasă în diferite contexte [22]. De exemplu, semiotica poate fi aplicată pentru a analiza campaniile publicitare, examinând simbolurile, semnele și narațiunile folosite pentru a influența consumatorii [120]. În literatură, semiotica ajută la descoperirea structurilor subiacente și a sistemelor simbolice din cadrul unui text, aruncând lumină asupra intențiilor autorilor și interpretărilor cititorilor [35]. Mai mult, semiotica își găsește aplicații în analiza artelor vizuale, a filmului, a muzicii și a comunicării non-verbale, oferind o perspectivă asupra modului în care este creat sensul prin diverse moduri de exprimare [12].

Integrarea potențială a semioticii și a învățării aprofundate reprezintă o cale interesantă de cercetare și explorare. Semiotica, cu accentul pus pe semne, simboluri și semnificație, oferă un cadru conceptual care poate îmbogăți și informa dezvoltarea și interpretarea modelelor de învățare aprofundată.

Învățarea profundă, un subdomeniu al inteligenței artificiale, presupune antrenarea rețelelor neuronale pentru a învăța și a extrage reprezentări semnificative din seturi mari de date [34]. Ea a obținut un succes remarcabil în sarcini precum recunoașterea imaginilor [53], procesarea limbajului natural [21] și sinteza vorbirii [112]. Cu toate acestea, una dintre provocările învățării profunde constă în capacitatea de interpretare a modelelor sale. Înțelegerea motivelor care stau la baza unei decizii sau a unei predicții specifice făcute de un sistem de învățare profundă poate fi o provocare din cauza complexității și a opacității algoritmilor de bază [39].

Prin încorporarea semioticii în procesul de învățare profundă, cercetătorii pot introduce un strat de interpretabilitate și semnificație în reprezentările învățate. Semiotica oferă o abordare structurată a analizei și interpretării semnelor și simbolurilor într-un context specific, aruncând lumină asupra reprezentărilor interne și a proceselor decizionale ale modelelor de învățare profundă. Ea oferă un cadru pentru a descoperi semnificațiile încorporate în date și pentru a stabili conexiuni între diferite caracteristici sau concepte.

O aplicație potențială a integrării semioticii și a învățării profunde este în domeniul vederii computerizate. Deși modelele de învățare profundă au demonstrat abilități remarcabile în recunoașterea și clasificarea obiectelor vizuale, interpretarea deciziilor lor rămâne o provocare. Prin încorporarea analizei semiotice, cercetătorii pot trece dincolo de identificarea obiectelor și pot explora semnificațiile simbolice și culturale asociate cu conținutul vizual. Această integrare poate dezvălui straturile implicite de semnificație din imagini, permițând interpretări mai nuanțate și îmbunătățind potențial performanța modelelor de învățare profundă în sarcini precum înțelegerea și generarea de imagini.

În plus, integrarea semioticii și a învățării profunde poate avea implicații practice pen-

tru proiectarea sistemelor centrate pe utilizator. Prin încorporarea analizei semiotice în procesele de instruire și de luare a deciziilor modelelor de învățare profundă, devine posibilă luarea în considerare a aspectelor culturale, sociale și contextuale ale comunicării umane. Această integrare poate duce la dezvoltarea unor sisteme mai incluzive și mai conștiente de context, care iau în considerare diversele semnificații și interpretări asociate cu semnele și simbolurile. Astfel de sisteme pot răspunde mai bine nevoilor și preferințelor utilizatorilor din medii culturale diferite, promovând o experiență mai incluzivă și mai ușor de utilizat.

1.5 Blocajul Informațional

Blocajul informațional (Information Bottleneck (IB)) este un cadru utilizat în învățarea automată și în teoria informației pentru a identifica și extrage informații relevante dintr-un set de date complexe, eliminând în același timp părțile redundante și irelevante. Ideea din spatele principiului IB este de a găsi o reprezentare comprimată a datelor de intrare care să rețină cât mai multe informații relevante posibil, minimizând în același timp cantitatea de zgomot și redundanță. Cu alte cuvinte, principiul IB urmărește să găsească un echilibru între compresie și acuratețe care să permită o procesare eficientă și eficace a informațiilor. Cadrul IB și-a găsit o utilizare practică în diverse domenii de cercetare, permițând cercetătorilor să extragă informații relevante din seturi complexe de date și să îmbunătățească eficiența procesării informațiilor. În ultimii ani, IB a câștigat o atenție tot mai mare ca abordare puternică pentru a aborda provocările legate de modelarea și luarea deciziilor bazate pe date.

Formularea originală a conceptului de IB a fost elaborată în [110] ca o tehnică teoretică a informației al cărei scop este de a găsi cel mai bun compromis între precizia predicției unei variabile Y și comprimarea variabilei aleatoare de intrare X în codul T. Acest lucru se realizează prin minimizarea următorului lagrangian [110]:

$$\min_{P_{T|X}} I(X;T) - \beta I(Y;T) \tag{1.1}$$

unde $I(\cdot,\cdot)$ este informația reciprocă a două variabile aleatoare, iar β este un parametru de compromis.

Recent, principiul IB a fost aplicat la învățarea profundă și prezentat de Tishby și Zaslavsky [111], ca un concept teoretic menit să ofere o posibilă explicație pentru mecanismele de bază care guvernează arhitecturile moderne de învățare profundă. Mecanismul din spate este similar cu cel din formularea originală, optimizând o reprezentare latentă T care reprezintă o statistică minimă suficientă pentru o intrare X, prin comprimarea oricăror informații redundante despre aceasta, păstrând în același timp informațiile necesare pentru a prezice eticheta Y. Acest lucru se face în același mod ca în ecuația 1.1. Lucrarea lor propune, de asemenea, limite teoretice privind capacitatea de gener-

alizare a unei rețele neuronale. Se susține că o bună generalizare este cauzată de o bună comprimare a intrării X în variabila latentă T. Principiul IB sugerează că straturile mai adânci corespund unor valori mai mici ale informației reciproce, oferind statistici din ce în ce mai comprimate [32]. Este important de remarcat faptul că autorii nu furnizează niciun experiment de instruire în care să utilizeze formularea IB.

Shwartz-Ziv și Tishby [93] au privit straturile unui DNN ca pe un lanț Markov de reprezentări interne succesive ale intrării X. Orice reprezentare latentă T este definită prin utilizarea unui codificator $P(T|X)$ și a unui decodificator $P(\hat{Y}|T)$, unde $\hat{Y}$ este predicția neuronală. Aceștia au definit noțiunea de plan informațional (IP) ca fiind planul de coordonate al cantităților de informații reciproce $I_X = I(X;T)$ și $I_Y = I(T;Y)$ pe parcursul mai multor perioade de instruire. Pentru un perceptron multistrat cu câteva straturi, antrenat pe o problemă de date sintetice, aceștia au observat două faze importante în timpul antrenamentului: o fază de ajustare, în care $I(X;T)$ și $I(T;Y)$ cresc ambele, și o fază de compresie, în care informația reciprocă $I(X;T)$ începe să scadă, în timp ce $I(T;Y)$ rămâne în mare parte constantă. Aceștia au asociat scăderea lui $I(X;T)$ cu comprimarea intrării X în latentul T, ceea ce evită supraadaptarea, explicând astfel generalizarea bună realizată de rețelele neuronale profunde (DNN) supraparametrizate.

Saxe *et al.* [89] a criticat ipoteza IP a lui Shwartz-Ziv și Tishby, argumentând că nu este aplicabilă la DNN-urile generale. Aceștia au susținut că cele două faze observate în [93] sunt cauzate de natura de saturație pe două fețe a funcției de activare *tanh* utilizată în MLP-ul lor și de clasificarea activărilor continue în valori discrete. Comportamentul bifazic, așa cum au demonstrat ei în mod empiric, nu este prezent în rețelele care utilizează funcții de activare nesaturate (cum ar fi ReLU), utilizate de majoritatea DNN-urilor moderne. Ei au testat, de asemenea, presupusa legătură dintre compresie și generalizare folosind rețeaua din [93], dar au antrenat-o pe un procent mai mic de date, arătând că, chiar dacă faza de compresie este vizibilă în PI, precizia de formare vs. testare suferă de o supraadaptare severă.

Wickstrøm *et al.* [119] au realizat primul experiment la scară largă folosind principiul IB, studiind arhitectura VGG16 [97] antrenată pe CIFAR-10 [52]. Aceștia au propus o entropie a lui Rényi bazată pe matrice, cuplată cu nuclee tensoriale peste straturile convoluționale pentru a estima informația mutuală dificil de rezolvat, pentru a analiza PI. Folosind această metodă, nu mai este nevoie de operații de binning. Una dintre observațiile lor a fost că compresia apare mai ales pe datele de instruire și este mai puțin vizibilă în setul de date de testare. În continuare, aceștia au utilizat un criteriu de oprire timpurie bazat pe un parametru de răbdare. Instruirea se oprește dacă precizia validării nu se modifică după un număr predefinit de epoci. Aceștia au observat că antrenamentul poate fi oprit uneori chiar înainte de începerea fazei de compresie. Se presupune aici că comprimarea este legată de supraadaptare.

Alte analize cuprinzătoare privind aplicațiile teoriei IB pot fi găsite în [28, 32]. Unele dintre concluziile desprinse din aceste analize sunt că IB trebuie explorată în continuare.

Compresia observată în IP-uri nu reprezintă neapărat învățarea unei statistici minime suficiente și nici faptul că produce o bună generalizare. Cu toate acestea, ea poate oferi o bună explicație geometrică pentru unele dintre comportamentele inerente care stau la baza DNN-urilor și ar putea chiar deschide ușile pentru înțelegeri teoretice mai profunde.

Pe măsură ce învățarea automată continuă să avanseze, este de așteptat ca cadrul IB să devină și mai relevant, mai ales că nevoia de modele de învățare automată eficiente și interpretabile continuă să crească. Prin urmare, abordarea IB va continua probabil să joace un rol crucial în îmbunătățirea performanței și a interpretabilității modelelor moderne de învățare automată.

1.6 Simplificarea Rețelelor Neuronale (Pruning)

Pruning-ul este o tehnică utilizată pentru a reduce numărul total de operații în virgulă mobilă pe secundă (FLOPS) și de parametri într-o rețea neuronală prin eliminarea ponderilor redundante. Această metodă este utilizată în mod obișnuit pentru a optimiza eficiența computațională a modelelor de învățare profundă, în special în mediile limitate din punct de vedere hardware.

Primele abordări pentru pruning neuronal au apărut în anii '90, cu metodele clasice de optimizare a leziunilor cerebrale [55] și de optimizare a chirurgului cerebral [41]. De atunci, s-a observat importanța pruning-ului în îmbunătățirea timpului de instruire și de inferență, o generalizare mai bună. Odată cu apariția rețelelor neuronale profunde de mari dimensiuni, pruning-ul a devenit și mai relevant și mai dezirabil, deoarece arhitecturile moderne ale rețelelor sunt supraparametrizate și există mult spațiu pentru optimizare. Ca atare, în ultimii ani a fost propusă o suită mare de metode de tăiere. Studii cuprinzătoare privind pruningul rețelelor neuronale pot fi găsite în [10, 26].

Învățarea automată a mașinilor (AutoML) a devenit din ce în ce mai importantă în dezvoltarea unor arhitecturi eficiente de rețele neuronale. Unul dintre principalele motive pentru acest lucru este faptul că spațiul de căutare pentru arhitecturile rețelelor neuronale este incredibil de vast, iar procesul de căutare manuală a unei arhitecturi optime poate fi consumator de timp și de resurse. Tehnicile AutoML pot contribui la automatizarea acestui proces, permițând cercetătorilor și practicienilor să exploreze în mod eficient o gamă mai largă de arhitecturi de rețea și hiperparametri [44].

O constatare cheie care stă la baza uneia dintre lucrările noastre este cea a lui He *et al.* [45]. În studiul lor, ei au utilizat un agent Deep Deterministic Policy Gradient (DDPG) [58] pentru a determina nivelul optim de raritate pentru fiecare strat învățabil, indiferent dacă era convoluțional sau complet conectat, prin maximizarea acurateței rețelei post-pruning. În cazul straturilor convoluționale de pruning, au identificat și au sparsificat filtrele cu cea mai mică magnitudine totală, în timp ce pentru straturile complet conectate, au eliminat cele mai mici greutăți. Această metodă de sparsificare

este cunoscută ca fiind structurată, deoarece greutățile sunt eliminate într-o structură specifică, cum ar fi un filtru întreg, mai degrabă decât să fie eliminate la întâmplare în cadrul unui filtru. Sparsificarea structurată este mai eficientă, deoarece filtre întregi pot fi eliminate din calcule, în timp ce sparsificarea aleatorie necesită efectuarea unor calcule în cadrul filtrelor și a altora nu, ceea ce complică logica aritmetică finală.

În ciuda numeroaselor sale beneficii, curățarea nu este lipsită de provocări. De exemplu, poate fi dificil să se determine ce ponderi, filtre sau neuroni trebuie să se elimine, iar diferite metode de eliminare pot avea rezultate diferite în funcție de arhitectură și sarcină. Mai mult, pruning-ul poate duce, de asemenea, la o pierdere de informații, ceea ce poate afecta acuratețea dacă nu se face cu atenție [25].

Cu toate acestea, pruning-ul a devenit o tehnică crucială în domeniul învățării profunde, permițând crearea unor modele mai eficiente, mai rapide și mai ecologice. Având în vedere progresele continue în materie de hardware și cererea tot mai mare de soluții de învățare automată, este probabil ca pruning-ul să rămână un domeniu vital de cercetare și dezvoltare în anii următori. Ca atare, îmbunătățirile suplimentare ale algoritmilor și tehnicilor de pruning vor continua, fără îndoială, să fie un domeniu de cercetare activă.

1.7 Context Matematic - Entropia Spațială

Atunci când se analizează rețelele neuronale convoluționale, este esențial să se ia în considerare entropia spațială a caracteristicilor convoluționale, mai degrabă decât să se bazeze doar pe calcule simple de entropie. CNN-urile operează cu date cu structuri spațiale bogate, cum ar fi imaginile sau datele corelate spațial. În aceste cazuri, calculele tradiționale ale entropiei pot trece cu vederea relațiile spațiale din cadrul datelor, ceea ce duce la reprezentări incomplete sau inexacte ale conținutului informațional. Prin calcularea entropiei spațiale a caracteristicilor convoluționale, putem surprinde dependențele și corelațiile spațiale care sunt esențiale pentru înțelegerea structurii de bază a datelor. Această entropie spațială oferă informații valoroase despre distribuția spațială a informațiilor în cadrul CNN-urilor, permițându-ne să luăm decizii mai bine informate în ceea ce privește comprimarea modelului, selecția caracteristicilor și optimizarea rețelei. Accentuând importanța analizei entropiei spațiale în cadrul CNN-urilor, ne putem îmbunătăți înțelegerea acestor modele complexe și putem valorifica aceste cunoștințe pentru a îmbunătăți performanța și interpretabilitatea.

De-a lungul tezei, folosim entropia matricei de aură spațială, așa cum a fost definită inițial în [114]. În analiza noastră, examinăm o grilă bidimensională, notată ca X, care poate fi extinsă și la trei dimensiuni în cazul unei hărți de caracteristici convoluționale care include mai multe canale. Definim probabilitatea comună a două celule de caracteristici în locațiile spațiale (i, j) și $(i + k,\ j + l)$ pentru a lua valorile g, respectiv g' ca:

$$p_{gg'}(k,l) = P(X_{i,\,j} = g, X_{i+k,\,j+l} = g') \tag{1.2}$$

unde g și g' sunt variabile discretizate, obținute după repartizarea valorilor hărților de acțiune. Dacă presupunem că $p_{gg'}$ este independent de $(i,\ j)$ (ipoteza de omogenitate [49]), definim pentru fiecare pereche $(k,\ l)$ entropia

$$H(k,\ l) = -\sum_{g}\sum_{g'} p_{gg'}(k,\ l)\log p_{gg'}(k,\ l) \tag{1.3}$$

unde sumele se raportează la numărul de valori posibile ale binelor. O măsură relativă standardizată a entropiei bivariate este [49]:

$$H_R(k,\ l) = \frac{H(k,\ l) - H(0)}{H(0)} \in [0,\ 1] \tag{1.4}$$

Entropia maximă $H_R(k,\ l) = 1$ corespunde cazului în care există două variabile independente. $H(0)$ este entropia univariată, care presupune că toate celulele caracteristice sunt independente și avem $H(k,\ l) \geq H(0)$.

Pe baza entropiei relative pentru $(k,\ l)$, Entropia de dezordine spațială (SDE) pentru o imagine $\mathbf{X}$ de m *orin* a fost definită în [49] ca:

$$H_{SDE}(\mathbf{X}) \approx \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{k=1}^{m}\sum_{l=1}^{n} H_R(i-k,\ j-l) \tag{1.5}$$

Deoarece complexitatea calculului SDE este mare, am decis să folosim o versiune simplificată - Aura Matrix Entropy (AME, a se vedea [114]), care ia în considerare doar vecinii de ordinul doi din calculul SDE:

$$H_{AME}(\mathbf{X}) \approx \frac{1}{4}\Big(H_R(-1,\ 0) + H_R(0,\ -1) + H_R(1,\ 0) + H_R(0,\ 1)\Big) \tag{1.6}$$

Punând totul cap la cap, pornind de la o hartă discretizată a caracteristicilor, calculăm probabilitățile $p_{gg'}$ din ecuația (1.2) și, în cele din urmă, AME din ecuația (1.6), care are ca rezultat cantitatea de entropie spațială a unei hărți de activare X.

Luarea în considerare a entropiei spațiale în analiza hărților de caracteristici și în calculul informației reciproce este de o importanță majoră, în special în domeniul rețelelor neuronale convoluționale (CNN). Prin luarea în considerare a caracteristicilor spațiale ale hărților de caracteristici, aprofundăm relațiile și modelele spațiale inerente datelor,

ceea ce conduce la reprezentări mai precise și mai semnificative. În plus, integrarea entropiei spațiale în calculul MI permite o evaluare cuprinzătoare a dependențelor statistice dintre variabile, permițând o mai bună înțelegere a relațiilor spațiale surprinse de CNN. Acest accent pus pe spațialitate sporește interpretabilitatea și eficiența CNN-urilor, facilitând o modelare și o analiză îmbunătățită în diverse domenii, cum ar fi viziunea computerizată, recunoașterea imaginilor și prelucrarea datelor spațiale.

Capitolul 2

Semne și Suprasemne în Modelele Neuronale Profunde

2.1 Agregarea Semiotică ca și Concentrare de Informații în Învățarea Profundă

Această secțiune se bazează pe lucrarea noastră publicată [73]. Textul original reprodus aici face parte din lucrarea noastră și nu este în niciun caz destinat plagiatului sau utilizării fără o atribuire corespunzătoare.

2.1.1 Motivația cercetării

Rețelele neuronale convoluționale au fost popularizate pentru prima dată de Yann LeCun *et al.* [54] cu lucrarea lor fundamentală privind recunoașterea caracterelor scrise de mână, unde au introdus arhitectura LeNet-5, populară în prezent. La acea vreme, constrângerile legate de puterea de calcul și lipsa de date au împiedicat aceste CNN-uri să își atingă adevăratul potențial în ceea ce privește capacitățile de viziune computerizată. Ani mai târziu, Krizhevsky *et. al* [53] au marcat începutul actualei revoluții a învățării profunde, când, în timpul competiției ILSVRC 2012, CNN-ul lor, intitulat AlexNet, și-a depășit concurentul din anul precedent cu o marjă de aproape 10%. De atunci, cercetările privind noile arhitecturi CNN au devenit foarte populare, producând candidați precum VGG [97], GoogleNet [104], ResNet [42] și, mai recent, EfficientNet [107].

În ciuda capacității de a genera predicții asemănătoare cu cele umane, CNN-urilor le lipsește încă o componentă majoră: interpretabilitatea. Rețelele neuronale, în general, sunt cunoscute pentru comportamentul lor de tip "cutie neagră", fiind capabile să capteze informații semantice folosind calcule numerice și învățarea bazată pe gradient, dar ascunzând mecanismele interne de funcționare a raționamentului. Cu toate acestea,

raționamentul este de o importanță crucială pentru domenii precum medicina, dreptul, finanțele, unde majoritatea deciziilor trebuie să fie însoțite de explicații bune pentru a lua o anumită acțiune în favoarea alteia. De obicei, există un compromis între acuratețe și interpretabilitate. De exemplu, regulile IF-THEN extrase dintr-o rețea neuronală sunt foarte ușor de interpretat, dar mai puțin precise.

De la apariția învățării profunde, s-au depus eforturi pentru a analiza problema interpretabilității și pentru a veni cu soluții potențiale care ar putea dota rețelele neuronale cu un simț al cauzalității [65,91,96,98,99,126,130]. Complexitatea ridicată a modelelor profunde face ca aceste modele să fie greu de interpretat. Nu este fezabil să extragem (și să interpretăm) regulile clasice IF-THEN dintr-un ResNet cu peste 200 de straturi.

Avem nevoie de diferite metode de interpretare pentru modelele profunde, iar o idee provine din procesarea/înțelegerea imaginilor. O tehnică obișnuită pentru înțelegerea deciziilor sistemelor de clasificare a imaginilor este de a găsi regiuni ale unei imagini de intrare care au avut o influență deosebită asupra clasificării finale. Această tehnică este cunoscută sub diferite denumiri: hartă de sensibilitate, hartă de salienț sau hartă de atribuire a pixelilor. Noi vom folosi termenul *mapă de salienț*. Hărțile de salienț sunt prezente și utilizate de mult timp în recunoașterea imaginilor. În esență, o hartă de salienț este o hartă topologică 2D care indică prioritățile atenției vizuale. Printre aplicațiile hărților de salienț se numără segmentarea imaginilor, detectarea obiectelor, reorientarea imaginilor, compresia imaginilor/video și designul publicitar [65].

Recent, hărțile de salienț au devenit un instrument popular pentru a obține informații despre învățarea profundă. În acest caz, hărțile de salienț sunt în mod obișnuit redate sub formă de hărți termice ale straturilor neuronale, în care "fierbințeala" corespunde regiunilor care au un impact mare asupra deciziei finale a modelului. Exemplificăm cu o abordare intuitivă bazată pe gradient, algoritmul Vanilla Gradient [96], care procedează după cum urmează: trecere înainte cu date, trecere înapoi la stratul de intrare pentru a obține gradientul, redarea gradientului ca hartă termică normalizată.

Cu siguranță, hărțile de salienț nu sunt instrumentul universal pentru interpretarea modelelor neuronale. Ele se concentrează asupra datelor de intrare și pot neglija să explice modul în care modelul ia decizii. Este posibil ca hărțile de salienț să fie extrem de asemănătoare pentru predicții de ieșire foarte diferite ale modelului neuronal. Un exemplu a fost oferit de Alvin Wan[1] folosind generatorul de hărți de salienț Grad-CAM (Gradient-weighted Class Activation Mapping) [91]. În plus, unele metode de salienț utilizate pe scară largă sunt incapabile să susțină sarcini care necesită explicații fidele modelului sau procesului de generare a datelor. Faptul de a se baza doar pe evaluarea vizuală a hărților de salienț poate fi înșelător și două teste pentru evaluarea domeniului de aplicare și a calității metodelor de explicare au fost introduse în [1].

O interpretare vizuală bună ar trebui să fie discriminatorie din punct de vedere al clasei (adică să localizeze categoria în imagine) și de înaltă rezoluție (adică să capteze detalii

[1]https://bair.berkeley.edu/blog/2020/04/23/decisions/#fn:saliency

de finețe) [91]. Guided Grad-CAM [91] este un exemplu de vizualizare care este atât de înaltă rezoluție, cât și discriminatorie în funcție de clasă: regiunile importante ale imaginii care corespund oricărei decizii de interes sunt vizualizate în detalii de înaltă rezoluție, chiar dacă imaginea conține dovezi pentru mai multe concepte posibile.

În abordarea noastră, ne concentrăm asupra aspectelor statistice ale proceselor de concentrare a informațiilor care apar în hărțile de salienţă ale straturilor CNN succesive. Analizăm hărțile de salienţă ale acestor straturi din perspectiva semioticii. În semiotica computațională, această operațiune de agregare (cunoscută sub numele de superizare) este însoțită de o scădere a entropiei spațiale: semnele sunt agregate în supersemne. O hartă de salienţă agregă informații din stratul anterior al rețelei. În semiotica computațională, această operațiune de agregare este cunoscută sub numele de *superizare* și poate fi măsurată printr-o scădere a entropiei spațiale. În acest caz, semnele sunt sintetizate în supersemne.

Contribuția noastră este originală și, după cunoștințele noastre, este prima aplicație a semioticii computaționale în analiza și interpretarea rețelelor neuronale profunde. *Semiotica* este cunoscută ca fiind studiul semnelor și al comportamentului de utilizare a semnelor. Conform [37], *semiotica computațională* este un domeniu interdisciplinar care propune un nou tip de abordare a sistemelor inteligente, în care o explicație explicită a noțiunii de semn este proeminentă. În lucrarea noastră, definiția semioticii computaționale se referă la aplicarea semioticii la inteligența artificială. Noi punem în slujba noțiunii de semn din semiotică pentru a oferi o nouă interpretare a învățării profunde, iar acest lucru este nou. Folosim conceptele semioticii computaționale pentru a explica procesele decizionale în modelele CNN. De asemenea, studiem posibilitatea de a aplica instrumente semiotice pentru a optimiza arhitectura rețelelor neuronale de învățare profundă. În prezent, optimizarea arhitecturii modelelor este un subiect de cercetare fierbinte în învățarea automată.

Intrările pentru modelul nostru sunt hărți de salienţă, generate pentru fiecare strat CNN prin Grad-CAM, care în prezent este o metodă de ultimă generație. Noi calculăm entropia hărților de salienţă, ceea ce înseamnă că cuantificăm conținutul informațional al acestor hărți. Acest lucru ne permite să studiem procesele de suprapunere care au loc între straturile succesive ale rețelei. În experimentele noastre, arătăm cum cunoștințele obținute pot fi utilizate pentru a explica modelul neuronal de decizie. În plus, încercăm să optimizăm arhitectura modelului neuronal utilizând o tehnică semiotică greedy.

2.1.2 Agregarea Semiotică și Teoria Informației în Învățarea Profundă

În această secțiune, obiectivul nostru este de a prezenta un cadru semiotic pentru analiza reprezentărilor vizuale, în special a hărților de salienţă, generate de rețele neuronale multistrat. Punctul central al abordării noastre constă în operațiunea de agregare pe straturi utilizată în cadrul acestor rețele. În timp ce teoria informației servește drept

instrument fundamental de calcul, aplicarea operației de agregare într-un cadru semiotic distinge munca noastră ca fiind o contribuție interdisciplinară.

În semiotică (sau *semiosis*), un *semn* este orice lucru care comunică un înțeles, care nu este semnul în sine, interpretului semnului. Această definiție este foarte generală. Definiții alternative aprofundate pot fi găsite în [13, 22, 90]. Noi luăm în considerare modelul triadic al semiozei, așa cum a fost enunțat de Charles Sanders Peirce. Peirce a definit semioza ca fiind o relație triadică ireductibilă între Semn-Obiect-Interpretant [79].

Charles Morris [69] a definit semiotica ca fiind grupată în trei ramuri:

- Sintactica: relații între sau între semne în structuri formale fără a ține cont de semnificație.
- Semantică: relația dintre semne și lucrurile la care se referă; denotația lor semnificată, sau semnificația.
- Pragmatică: relațiile dintre sistemul de semne și utilizatorul său uman (sau animal).

Într-o manieră simplistă, semiotica a jucat deja un anumit rol în informatică în anii '60. Distincția dintre sintactică, semantică și pragmatică făcută de Charles Morris a fost la acea vreme importată în teoria limbajelor de programare [127]. Rezultate mai recente pot fi găsite în [108].

Semiotica computațională se bazează pe o descriere matematică a conceptelor din semiotica clasică. În [36], se afirmă că rețelele semantice pot implementa modele de inteligență computațională: sisteme fuzzy, rețele neuronale și algoritmi de calcul evolutiv. Ulterior, au fost propuse unele modele computaționale ale noțiunii triadice de procese de semnificație a lui Peirce [33, 37, 38].

În această lucrare, ne concentrăm asupra aspectelor computaționale ale semioticii în învățarea profundă. Infrastructura noastră semiotică se află la intersecția dintre teoria lui Peirce și teoria informației, o teorie dezvoltată de Max Bense [8] și Helmar Frank [24].

Semnele obișnuite desemnează entități materiale care sunt percepute inconștient. Aceste așa-numite *semne de prim nivel* pot fi aglomerate în semne de la nivelul ierarhic următor, numite *semne superioare de al doilea nivel*. Iterând procesul, obținem *k-lea nivel de supersemne* mai abstracte. Tranziția de la al k-lea nivel la $(k+1)$-lea nivel de supersemne se numește *superizare*. Frank [24] a identificat două tipuri de superizare:

1. **Tipul I** "Durch Klassesbildung" (prin formarea de clase, în germană): construirea de clase de echivalență și astfel reducerea numărului de semne. Literele unui text pot fi considerate semne de prim nivel. Clasa de echivalență a tuturor tipurilor de literă "a" (scrisă de mână, majusculă și așa mai departe) este un supersemn de al doilea nivel.

2. **Tipul II** "Durch Komplexbildung" (prin formare compusă, în germană): construirea de supersemne compuse din supersemne componente mai simple. Reconsiderând exemplul anterior, putem obține în acest fel cuvinte din litere, propoziții din cuvinte și, ulterior, structuri sintactico-semantice din ce în ce mai complexe și mai abstracte.

Superizarea este un proces de agregare semiotică caracterizat la fiecare nivel de percepție de un repertoriu specific de supersemne. Structurile ierarhice de date de viziune computerizată (de exemplu, quadtrees, piramide cu rezoluție multiplă) pot fi considerate superizări simpliste [3, 4]. Ideea de bază este de a trata fiecare componentă ca un pixel la nivelul ierarhic dat. În acest caz, există o asemănare între procesele de reprezentare agregată ierarhică și cele de superizare. Cu toate acestea, există și diferențe: superizările nu sunt procese combinatorii simple, ci cadre subtile de percepție sintactico-semantică legate de modelul triadic al semiozei lui Peirce.

O reprezentare a imaginii cu mai multe rezoluții poate fi caracterizată la fiecare nivel printr-o măsură a informației. Entropia Shannon dependentă de rezoluție poate fi derivată din distribuția de probabilitate a evenimentelor de nivel de gri observate la nivelul respectiv [121]. Folosind analogia cu citirea ziarului, la nivelul mărit, unde sunt vizibile doar pete albe și negre, entropia H va fi scăzută. Pe măsură ce imaginea este adusă la distanța normală de focalizare, o mare varietate de niveluri de gri devin vizibile și, în consecință, entropia crește. Pe măsură ce imaginea este îndepărtată mai mult de ochi, entropia scade. În cele din urmă, imaginea poate avea un aspect gri aproape uniform, cu $H \approx 0$. Observația care se asociază cu valoarea maximă a entropiei este una dintre cele mai semnificative observații ale imaginii. Cu toate acestea, din cauza altor factori, entropia maximă nu este întotdeauna asociată cu rezoluția "optimă" [4].

Din punct de vedere al psihologiei informaționale, entropia crește până când atinge valoarea maximă. În opinia noastră [3, 4], această fază poate fi asociată cu adaptarea informațională a perceptorului. Scăderea ulterioară a entropiei este legată de procesarea informațiilor structurale [121]. Rata de scădere depinde în mare măsură de cantitatea de informații structurale din imagine. Entropia scade rapid atunci când sunt disponibile puține informații structurale, în timp ce atunci când sunt prezente informații structurale importante, entropia va rămâne ridicată pe cea mai mare parte a intervalului său. Variația entropiei poate indica tipul și cantitatea de informații structurale din imagine în ceea ce privește dimensiunea și relațiile cu caracteristicile detaliate. În studiul actual, ne concentrăm doar pe faza de scădere a entropiei, deoarece CNN-urile analizate nu se adaptează la intrări prin modificarea dinamică a rezoluției imaginii de intrare.

Ideea de a considera straturile CNN ca reprezentări multi-rezoluție ale imaginilor de intrare este interesantă, dar nu foarte nouă [14, 43, 50]. De exemplu, în [43] se introduce un strat de grupare a piramidei spațiale între straturile convoluționale și straturile complet conectate, pentru a evita necesitatea decupării sau deformării imaginilor de intrare. În [14], straturile de convoluție de intrare la mai multe rate de eșantionare sunt aplicate straturilor convoluționale pentru a capta obiectele, precum și contextul

imaginii la mai multe scări.

În abordarea noastră, considerăm exemplul de reprezentare a imaginilor cu rezoluție multiplă în contextul unui proces de recunoaștere semiotică, în care mașina (sau interpretul) încearcă să clasifice o imagine de intrare. Ne imaginăm procesul de recunoaștere ca pe un clasificator neuronal multistrat de tip feedforward, în care fiecare strat realizează o suprapunere a stratului anterior. Presupunem că informația subiectivă (măsurată prin entropie) este pusă la dispoziția unui interpret (de exemplu, calculatorul sau supraveghetorul uman) care încearcă să clasifice imaginea de intrare.

Să luăm în considerare entropiile calculate în două straturi succesive: H_k și H_{k+1}. Informația extrasă de către interpret poate fi măsurată prin diferența $H_k - H_{k+1}$. Detalii pot fi găsite în [100]. Avem următorul rezultat:

Theorem 2.1. *(din [24]): Superizarea tinde să concentreze informația prin scăderea entropiei.*

Proof
Considerăm separat cele două tipuri de superizare. Pentru un set $Z = (Z_1, \ldots, Z_n)$ de supersemne cu probabilitățile corespunzătoare $p_1, \ldots, p_n$, $\sum p_i = 1$, folosind o superizare de primul tip, putem obține supersemne de nivelul următor $Z^* = (Z_1, \ldots, Z_{n-2}, \{Z_{n-1}, Z_n\})$ cu probabilitățile corespunzătoare $p_1, \ldots, p_{n-2}, p_{n-1} + p_n$. Avem următoarea inegalitate: $H(Z) = \sum p_i \, log \, p_i \geq H(Z^*)$.

□

Pentru două seturi de suprasemne X și Y, folosind al doilea tip de superizare, se obțin suprasemne compuse din setul comun $Z = (X, Y)$. O relație bine cunoscută completează dovada: $H(X) + H(Y) \geq H(Z)$.

O aplicație intuitivă a acestei teoreme este atunci când luăm în considerare straturile neuronale ale unui CNN. O suprapunere de tip I apare atunci când reducem rezoluția spațială a unui strat $k + 1$ prin subeșantionarea stratului k. Acest lucru este similar cu formarea claselor, deoarece reducem variația valorilor de intrare (adică reducem numărul de semne). În cazul CNN-urilor, acest lucru se realizează de obicei printr-un operator de grupare. Operatorul de grupare poate fi considerat ca o formă de eșantionare descendentă neliniară care împarte imaginea de intrare într-un set de dreptunghiuri care nu se suprapun și, pentru fiecare subregiune, calculează media (grupare medie) sau valoarea maximă (grupare maximă). Formula pentru gruparea maximă aplicată unei hărți de caracteristici F la nivelul k și la locațiile (i, j) cu un nucleu de 2 $ori 2$ este următoarea:

$$O_{i,j}(F) = max(F_{i,j}, F_{i+1,j}, F_{i,j+1}, F_{i+1,j+1}) \tag{2.1}$$

O suprapunere de tip II este produsă atunci când se aplică un operator convoluțional unui strat neuronal k. Ca efect, stratul $k + 1$ se va concentra pe obiecte mai complexe,

compuse din obiecte deja detectate de stratul k. Operatorul convoluțional pentru o hartă de caracteristici F la nivelul stratului k și locațiile pixelilor (i, j) cu un nucleu W de $3x3$ are următoarea formulă:

$$O_{i,j}(F) = \sum_{x=0}^{2} \sum_{y=0}^{2} F(i+x, i+y)W(x, y) \tag{2.2}$$

Ieșirea O a operatorului convoluțional este o combinație liniară a caracteristicilor de intrare și a ponderilor kernelului învățat. Astfel, un neuron rezultat va fi capabil să detecteze o combinație de obiecte mai simple care formează un obiect mai complex, prin compunerea de supersemne.

Observăm că efectul superizării este o tendință de scădere a entropiei la fiecare nivel. Acest lucru este diferit față de cazul reprezentării imaginilor cu mai multe rezoluții. În [4] am explicat această diferență prin următoarea teză: "Semnele de la primul nivel sunt percepute la un nivel de complexitate care corespunde rezoluției "optime"". Cu toate acestea, această teză nu se aplică unui model de recunoaștere computerizată (un clasificator), ci percepției umane.

Într-o formă simplificată, un clasificator cu mai multe niveluri poate fi interpretat din teoria semiotică a lui Morris ca o tranziție: sintactică-semantică-pragmatică. La finalul unui proces de recunoaștere reușit, entropia stratului de ieșire devine 0 și nu mai este nevoie să se extragă alte informații. Ultimul strat (stratul complet conectat într-o rețea CNN) este conectat la lumea exterioară, lumea obiectelor. Acesta poate fi considerat nivelul pragmatic în teoria semiotică a lui Morris, deoarece arată relația dintre semnele de intrare și obiectele de ieșire care pot fi legate de decizii și acțiuni.

2.1.3 Semne și Suprasemne în Hărțile de Salienț ă CNN

Teorema 1 este o simplificare a proceselor de suprapunere care au loc în straturile succesive ale hărților de salienț ă. Avem atât formarea claselor, cât și suprapunerea formelor compuse, iar entropia calculată este spațială. Calculăm superizările la nivelul hărților de salienț ă. Cu alte cuvinte, semnele și suprasemnele noastre se referă la valorile calculate în hărțile de salienț ă succesive calculate prin metoda Grad-CAM.

Ipoteza noastră este că, în centrul unui CNN, există ambele tipuri de superizări. În cazul superizării de tip I (prin formarea de clase), operația de punere în comun combină semnele (valori scalare) după criterii precum valoarea medie sau valoarea maximă, rezultând un singur semn, reducând astfel numărul acestora și construind clase de echivalență. O altă interpretare potențială a operației de punere în comun este că aceasta construiește clase de echivalență prin gruparea elementelor vecine din punct de vedere spațial. În experimentele noastre (după cum vom vedea în secțiunea 2.1.4), acest fenomen a putut fi observat după fiecare strat de grupare, unde magnitudinea entropiei spațiale a hărților de salienț ă ar avea o scădere mare. Din punct de vedere

vizual, hărțile de salienț㠠încep să se concentreze mai mult în jurul regiunilor conectate pe măsură ce se formează semne mai complexe.

Pentru superizarea de tip II (prin formarea de compuși), se știe că CNN compun obiecte întregi pornind de la părți simple de obiecte [126]. Acest fenomen descrie exact cel de-al doilea tip de superizare, deoarece construiește supersemne compuse pornind de la supersemnele componente mai simple. Ei reușesc să facă acest lucru prin mărirea treptată a câmpului receptiv după aplicarea fiecărui strat convoluțional. Pe măsură ce câmpul receptiv crește, un singur neuron din cadrul unui strat ascuns poate acoperi o regiune de interes mult mai mare din imaginea de intrare și, astfel, se activează pentru obiecte din ce în ce mai complexe.

Ceea ce complică interpretarea în cazul rețelelor CNN este faptul că, pentru unele straturi, ambele supraconvoluții operează simultan și poate fi dificil să se separe efectele lor.

Ipoteza noastră este că, pentru a scădea în mod vizibil entropia spațială, primul tip de superizare este mai eficient, în timp ce al doilea tip este mai responsabil cu construirea de supersemne cu roluri semantice, fără a afecta atât de mult entropia spațială.

2.1.4 Experimente și Discuții

Scopul următoarelor experimente este de a explora variația entropiei spațiale a hărților de salienț㠠calculate cu Grad-CAM pe câteva arhitecturi CNN reprezentative. Ne așteptăm ca entropia să scadă odată cu adâncimea și că acest lucru poate fi legat de procesele de suprapunere de tip I.

Luăm în considerare trei arhitecturi de rețea standard: AlexNet [53], VGG16 [97], și ResNet50 [42]. În plus, studiem, de asemenea, variația entropiei pe o rețea personalizată de tip LeNet-5[2].

Experimentele sunt efectuate în contexte diferite pe următoarele seturi de date:

1. Un subset de ImageNet [19] compus din clasa "castor" din setul de instruire, pentru a testa cazurile de utilizare preinstruite și inițializate aleatoriu.
2. CIFAR-10 [52] to: *a)* să antreneze rețeaua personalizată fără eșantionare redusă; și *b)* să testeze rețeaua nou-antrenată și una inițializată aleatoriu, cu aceeași arhitectură, folosind acest set de date ca set de testare.
3. "kangaroo" class from Caltech101 [57] pentru a testa o rețea preinstruită pe ImageNet. Faptul că instruim și testăm pe seturi de date diferite (dar oarecum similare) poate avea un impact asupra performanței de generalizare a rețelei și poate expune o posibilă supraadaptare la datele de instruire. Acest lucru este

[2]Originalul LeNet-5 a fost introdus în [54].

cunoscut sub numele de învățare *zero-shot* și poate fi privit ca un caz extrem de adaptare la domeniu.

4. Caltech101 [57] pentru a testa cazul în care rețeaua este preînvățată pe ImageNet, apoi antrenată (ajustată fin) pe Caltech101. Aceasta este abordarea *învățare prin transfer*.

Experimente pe Arhitecturi CNN Standard

Prezentăm rezultatele experimentale pentru fiecare dintre arhitecturile CNN luate în considerare. În tabelele următoare, folosim următorii termeni: (i) Preformate - greutăți preformate disponibile public pe ImageNet, (ii) Aleatoare - greutăți inițializate aleatoriu, (iii) Reglare fină - greutăți reglate fin pornind de la cele preformate și formate pe ImageNet, (iv) ImageNet - clasa "castor" din setul de formare ImageNet, (v) Caltech101 - clasa "cangur" din setul de formare Caltech101.

AlexNet [53] este compus dintr-o secvență de straturi convoluționale, de max-pooling și ReLU, urmate la final de straturi complet conectate care proiectează liniar caracteristicile extrase din coloana vertebrală convoluțională către numărul dorit de clase de ieșire. Tabelul 2.1 surprinde valorile rezultatelor experimentale pentru fiecare strat al rețelei.

AlexNet				
Strat	Pretrained ImageNet	Random ImageNet	Pretrained ImageNet	Fine-tuning Caltech101
conv1	0.6830	0.6816	0.6786	0.6829
relu1	0.6806	0.6802	0.6746	0.6795
maxpool1	0.5252	0.5113	0.5264	0.5356
conv2	0.5311	0.5100	0.5395	0.5352
relu2	0.5231	0.5096	0.5297	0.5191
maxpool2	0.4147	0.3952	0.4241	0.4116
conv3	0.4423	0.3861	0.4508	0.4474
relu3	0.4326	0.3864	0.4437	0.4454
conv4	0.4272	0.3867	0.4375	0.4292
relu4	0.4214	0.3934	0.4222	0.4304
conv5	0.4056	0.3934	0.4019	0.3925
relu5	0.3928	0.3949	0.3878	0.3784
maxpool3	0.3114	0.3038	0.3077	0.3071

Table 2.1: Valorile entropiei pentru hărțile de salienă pentru AlexNet la diferite niveluri din rețea. Tabel reprodus din [73].

VGG16 [97] are o arhitectură relativ simplă și compactă, constând în doar 3×3 convoluții, max-pooling și ReLU, urmate de mai multe straturi complet conectate. Șmecheria din spatele arhitecturii VGG16 constă în utilizarea a două convoluții

secvențiale de 3×3 pentru a înlocui o convoluție mai mare de 5×5, obținându-se astfel aceeași acoperire a câmpului receptiv prin utilizarea mai puțini parametri. Problema VGG16 este că majoritatea parametrilor săi se află în straturile complet conectate, ceea ce face ca rețeaua să fie foarte ineficientă din punct de vedere al parametrilor și al memoriei. Tabelul 2.2 prezintă valorile entropiei la diferite niveluri ale rețelei.

VGG16				
Strat ImageNet	Pretrained ImageNet	Random ImageNet	Pretrained Caltech101	Fine-tuning Caltech101
conv1	0.8516	0.785	0.8418	0.8369
conv3	0.8017	0.7322	0.7883	0.7731
conv5	0.6742	0.6308	0.6681	0.648
conv10	0.5491	0.5155	0.5556	0.5429
conv12	0.5112	0.5155	0.5127	0.4901
conv13	0.4213	0.4035	0.4281	0.4135
conv14	0.3868	0.4288	0.3994	0.3599
maxpool5	0.3131	0.3443	0.3238	0.3086

Table 2.2: Valorile entropiei pentru hărțile de salienț pentru VGG16 la diferite niveluri din rețea. Tabel reprodus din [73].

Noutatea ResNet [42] constă în conexiunile reziduale care atenuează problema gradientului de dispariție, o problemă care a urmărit rețelele neuronale profunde încă de la începuturile lor. În timpul retropropagării, gradienții ar începe să scadă treptat în magnitudine din cauza regulii lanțului aplicată la valori foarte mici, până când devin 0 și, în consecință, multe straturi ar fi lipsite de orice semnal de gradient pe baza căruia să își actualizeze ponderile respective. ResNet rezolvă această problemă prin crearea de ramuri reziduale de la un bloc de intrare la un bloc de ieșire sub forma $y = x + f(x)$, unde x este intrarea blocului și $f(x)$ este o secvență de straturi multiple. În loc să învețe o funcție, ca în cazul arhitecturilor anterioare, cum ar fi AlexNet sau VGG16, ResNets încearcă să învețe un reziduu pentru intrarea x, de unde și numele arhitecturii. Valorile entropiei pentru diferite straturi sunt prezentate în tabelul 2.3.

Pentru toate cele trei rețele observăm o tendință de scădere a entropiei spațiale, în special după straturile de max-pooling, care, în ipoteza noastră, sunt straturile responsabile de suprapunerea de tip I. Superizarea de tip II poate fi observată prin aplicarea mai multor straturi convoluționale consecutive. În acest caz, entropia spațială nu scade neapărat, dar scopul general este de a lărgi câmpul receptiv al rețelei, astfel încât neuronii să se activeze pentru obiecte mai complexe în timp ce progresează prin straturi.

Având în vedere experimentele noastre de mai sus și faptul bine cunoscut că CNN compun obiecte complexe pornind de la cele mai simple, acest lucru susține ipoteza noastră că suprapunerea de tip I este mai eficientă pentru scăderea entropiei. Nu am observat o scădere sistematică a entropiei în cazul suprapunerii de tip II și concluzionăm că aceasta este mai responsabilă pentru construirea de suprasemne cu roluri semantice.

ResNet50				
Strat	Pretrained ImageNet	Random ImageNet	Pretrained ImageNet	Fine-tuning Caltech101
conv1	0.7854	0.6574	0.7705	0.7633
block1	0.6849	0.5108	0.6807	0.6794
block2	0.5912	0.4193	0.5901	0.582
block3	0.4574	0.3398	0.4588	0.4607
block4	0.2847	0.3019	0.2754	0.2862

Table 2.3: Valorile entropiei pentru hărțile de salienț pentru ResNet50 la diferite niveluri din rețea. Tabel reprodus din [73].

Optimizarea Arhitecturii CNN

Este cunoscut faptul că arhitecturile moderne ale rețelelor neuronale sunt supraparametrizate [76] și, prin urmare, o tendință emergentă importantă în învățarea profundă este optimizarea acestor rețele neuronale profunde pentru a satisface diverse constrângeri hardware. O prezentare generală a acestor tehnici de optimizare poate fi găsită în [17, 103]. Dintre acestea, pruning-ul este considerat o metodă fundamentală, care a fost studiată încă de la sfârșitul anilor 1980 [70], și constă în reducerea operațiilor redundante prin eliminarea conexiunilor inutile sau slabe la nivel de greutăți sau straturi. În ultimii doi ani, metodele de pruning de ultimă generație au avansat considerabil și sunt acum capabile să reducă de câteva ori sarcina de calcul a unei rețele neuronale profunde, fără a suferi pierderi de precizie [10].

Experimentele descrise în secțiunea 2.1.4 au arătat că entropia spațială a hărților de salienț CNN scade, în general, strat cu strat, iar acest lucru poate fi legat de superizarea semiotică. Ne propunem să arătăm cum această interpretare ar putea contribui, de asemenea, la optimizarea (sau simplificarea) arhitecturii rețelei. Efectuăm un studiu de ablație pentru a vedea dacă putem determina straturile redundante pentru tăiere pe baza informațiilor privind entropia spațială a hărților de salienț. Este dincolo de scopul acestei lucrări să comparăm sistematic abordarea noastră cu alte tehnici de optimizare a arhitecturii CNN. Explorăm acest domeniu doar ca o dovadă de concept, deoarece este pentru prima dată când o astfel de metodă semiotică este utilizată pentru optimizarea arhitecturii neuronale.

Pe rețeaua VGG16, aplicăm în mod iterativ următorul algoritm greedy: (i) antrenăm rețeaua pe CIFAR-10 utilizând optimizatorul SGD cu o rată de învățare de 0.01; (ii) calculăm entropia spațială pentru fiecare hartă de salienț; (iii) eliminăm un strat pentru care entropia nu scade; și (iv) repetăm etapele (i)-(iii) cât timp performanța nu se degradează prea mult.

Din rezultatele obținute, observăm că până la 8 straturi convoluționale pot fi eliminate complet din rețea, iar acest lucru afectează performanța cu mai puțin de 1%. La eliminarea celui de-al 9-lea strat, precizia scade semnificativ; prin urmare, oprim procesul

iterativ în această etapă.

O constatare interesantă este aceea că ordinea în care eliminăm straturile are o importanță semnificativă. Dacă straturile mici cu parametri puțini de la începutul rețelei sunt eliminate primele, acuratețea scade cu 2% după a 3-a eliminare. Atunci când se elimină straturile mari (supraparametrizate) începând de la nivelul de mijloc al rețelei, precizia se menține. Precizia se degradează deosebit de rapid după eliminarea stratului convoluțional secund cu 64 de canale de ieșire.

Explicația noastră este că primele două straturi convoluționale sunt cruciale pentru performanța în aval a rețelei. Această primă parte a unei rețele, înainte de aplicarea unei operații de subeșantionare, este cunoscută în literatura de specialitate sub numele de stem [105]. Unele variante de ResNets implementează această tulpină sub forma a trei straturi convoluționale de 3×3 sau a unui strat mare de 7×7. Aceste straturi timpurii sunt responsabile de detectarea caracteristicilor de nivel scăzut, cum ar fi detectorii de margini. Având doar un singur strat convoluțional de 3×3, în loc de două sau trei, înseamnă că câmpul receptiv înainte de prima operație de max-pooling este de 3×3, ceea ce ar putea fi prea mic pentru a detecta în mod corespunzător trăsăturile și marginile de bază.

Rețeaua rezultată are următoarea configurație: 64, 64, M, 128, M, M, 256, M, M, 512, M, M, M, unde "M" reprezintă max-pooling, iar numerele întregi reprezintă un strat convoluțional cu numărul respectiv de canale de ieșire, urmat de o neliniaritate ReLU. Straturile complet conectate nu se modifică față de arhitectura originală. Comparăm rețeaua noastră rezultată cu VGG11, care este cea mai mică arhitectură din familia VGG. Rezultatele sunt afișate în tabelul 2.4. Se poate observa că, chiar și atunci când se reduce capacitatea rețelei cu un factor de aproximativ 7.5 $\times$, acuratețea se menține, ceea ce înseamnă că rețeaua este prea supraparametrizată pentru această sarcină.

Rețea	Numărul de parametri	Precizie
VGG16	15.245.130	89.55%
VGG11	9.750.922	87.83%
VGG16 after 4 layers removed	9.345.354	89.57%
VGG16 after 8 layers removed	2.118.346	89.49%

Table 2.4: Comparații pe CIFAR-10 - acuratețea de top 1 între VGG16, VGG11 (cea mai mică configurație din familia VGG), VGG16 după eliminarea a 4 straturi (care are aproximativ același număr de parametri ca VGG11) și VGG16 după eliminarea a 8 straturi (care este cea mai mică configurație care menține acuratețea la o diferență de 1%). Tabelul este reprodus din [73]

2.1.5 Concluzii

Am introdus o nouă interpretare semiotică computațională a arhitecturilor CNN, bazată pe aspectele statistice ale proceselor de concentrare a informațiilor (superizări semiotice) care apar în hărțile de salientă ale straturilor CNN succesive. În centrul unui CNN, cele două tipuri de superizări coexistă. Conform rezultatelor noastre, primul tip de superizare este eficient în ceea ce privește diminuarea entropiei spațiale. Tipul II de superizare este mai responsabil pentru construirea de supersemne cu roluri semantice.

Dincolo de aspectul exploratoriu al lucrării noastre, principalele noastre concluzii sunt de două feluri. Pe partea de extragere a cunoștințelor, interpretarea obținută poate fi utilizată pentru a vizualiza și explica procesele decizionale în cadrul modelelor CNN. Pe partea de optimizare a modelelor neuronale, întrebarea este cum să folosim informațiile semiotice extrase din hărțile de salientă pentru a optimiza arhitectura CNN-ului. Am reușit să simplificăm în mod semnificativ arhitectura unui CNN utilizând o tehnică semiotică greedy. Deși acest proces de optimizare poate fi lent, lucrarea noastră încearcă să utilizeze noțiunea de semiotică computațională pentru a curăța o rețea de ultimă generație existentă printr-o abordare de sus în jos, în loc să construiască una folosind o abordare de jos în sus, cum ar fi căutarea arhitecturii neuronale. În cadrul lucrărilor viitoare trebuie să se efectueze o analiză aprofundată pentru a lua în considerare alte arhitecturi de rețea și robustețea metodei.

Unele îmbunătățiri computaționale pentru calcularea entropiei spațiale au fost propuse de Razlighi *et al.* [81, 82]. Sarcina de calcul poate fi redusă semnificativ dacă acceptăm o reducere a preciziei aproximației. Intenționăm să folosim acest truc în viitor.

În această lucrare am luat în considerare doar un singur tip de topologie de rețea neuronală: CNN. Deoarece CNN-urile sunt potrivite mai ales pentru imagini, acestea au devenit subiectul studiului nostru. În viitor, intenționăm să studiem conexiunea cu alte domenii (audio, text) și tipuri de arhitectură (rețele neuronale recurente). Abordarea semiotică poate fi extinsă și la alte modele de învățare profundă, deoarece suprapunerea semiotică pare să fie prezentă în multe arhitecturi. Abordarea semiotică computațională este foarte promițătoare, în special pentru explicarea și optimizarea rețelelor profunde, în care sunt implicate mai multe niveluri de superizare.

2.2 O Interpretare Semiotică a Principiului Blocajului Informațional

Următoarea secțiune se bazează pe lucrarea noastră publicată [87]. Textul original reprodus aici face parte din lucrarea noastră și nu este în niciun caz destinat plagiatului sau utilizării fără atribuirea corespunzătoare.

2.2.1 Motivația cercetării

Rețelele neuronale profunde moderne sunt mașini de calcul capabile să reprezinte funcții foarte complexe, care pot rezolva o suită de sarcini extrem de dificile, de la viziunea computerizată [75], [115], la procesarea limbajului natural [125], [31] și robotică [74], [80]. Deși posedă o putere de expresie ridicată, interpretabilitatea modelelor a fost întotdeauna un factor limitativ pentru cazurile de utilizare care necesită explicații ale caracteristicilor implicate în modelare. Domeniul interpretabilității/explicabilității în învățarea profundă a cunoscut o explozie de lucrări publicate în ultimii ani. Chiar dacă nu există o teorie fundamentală care să poată elucida toate mecanismele subiacente prezente în aceste rețele, multiple lucrări au încercat să abordeze această problemă, venind cu soluții parțiale, fie prin explicații vizuale [126], [91], fie prin perspective teoretice [111], [93], [73]. Prin urmare, putem spune că interpretarea de tip "cutie neagră" a învățării profunde nu mai este adevărată și că avem nevoie de tehnici mai bune pentru a interpreta aceste modele. În cele ce urmează, ne vom referi la trei metode utilizate pentru interpretarea învățării profunde.

Motivația principală a lucrării noastre este de a testa ipoteza IB pe o varietate de situații noi, având în vedere că există opinii și rezultate contradictorii cu privire la teoria IB (a se vedea, de exemplu, [89]). Teza noastră este că există o asemănare semnificativă între principiul IB și superizarea semiotică. După cunoștințele noastre, acest aspect sinergetic nu a mai fost discutat până acum.

Investigăm teoria IB în contextul proceselor de superizare semiotică în învățarea CNN. În termeni practici, studiem evoluția entropiei spațiale a hărților de salienţă CNN pentru a valida/invalida principiul IB. În experimentele noastre, am observat un model similar cu fazele de ajustare și compresie care apar în evoluția entropiei spațiale. Folosim aceste rezultate experimentale pentru a antrena o rețea prin înghețarea straturilor redundante cu o variabilitate scăzută a entropiei spațiale. Acest lucru ne permite să descoperim analogii interesante între teoria IB și superizarea semiotică.

Contribuțiile noastre sunt duble: stabilim o legătură între ipoteza IB de ajustare și compresie și superizarea semiotică prin intermediul evoluției entropiei spațiale aplicate hărților de salienţă. Ca aplicație practică, proiectăm o strategie euristică de instruire pentru oprirea timpurie a straturilor pe baza variabilității entropiei spațiale în timp, care poate fi utilizată pentru a preveni supraadaptarea în timpul învățării.

2.2.2 Superizarea și Principiul Blocajului Informațional

Această secțiune prezintă teza noastră cu privire la analogia dintre adaptarea informației prin superizare și principiul IB.

Pe lângă superizare, este interesant de studiat și un alt aspect semiotic într-un model CNN - adaptarea informațională. Acest aspect nu a fost discutat niciodată până acum. În experimentele noastre preliminare, am observat că, în timpul antrenamentului, entropia fiecărui strat neuronal crește până când atinge valoarea maximă. Această fază poate fi asociată cu adaptarea informațională a modelului. Scăderea ulterioară a entropiei este legată de procesarea informațiilor structurale. Rata de scădere este în mare măsură dependentă de cantitatea de informații structurale din stratul de intrare. Atunci când există puține informații structurale, entropia scade rapid, în timp ce atunci când există elemente structurale majore, entropia straturilor neuronale rămâne ridicată pe cea mai mare parte a intervalului său. Modul în care se schimbă entropia indică tipul de informații din stratul de intrare [4]. În opinia noastră, această adaptare a informației poate fi legată de cele două faze distincte ale principiului IB - potrivire și compresie.

Pentru a explora prezența ipotezei IB în hărțile de salienă, investigăm:

- evoluția informației reciproce în timpul antrenamentului, între hărțile de salienă de intrare și cele intermediare și între hărțile de salienă intermediare și cele de ieșire.
- Evoluția entropiei spațiale pentru hărțile de salienă în timpul antrenamentului.

Analiza planului IB, așa cum este descrisă în [93], a urmărit cele două cantități de informație reciprocă $I(X;T)$ și $I(Y;T)$ și a observat apariția modelelor de ajustare și compresie. Ca atare, analizăm planurile de informații dintre $I(X;T)$ și $I(Y;T)$ prin calcularea informației reciproce din ecuația 3.1 între prima și o hartă de salienă intermediară, precum și între ultima și aceeași hartă de salienă intermediară. Experimentul propus este menit să descopere orice asemănare cu rezultatele originale din [93], dar aplicate la un concept diferit, cum ar fi hărțile de salienă.

Mergând puțin mai departe, testăm o posibilă legătură între modelele de ajustare și compresie prezente în teoria IB cu entropia spațială a hărților de salienă. În [73], entropia spațială a fost studiată într-un singur punct în timp (după antrenament), mergând de-a lungul adâncimii rețelei. Acum intenționăm să surprindem dinamica entropiei pe parcursul întregii instruiri pentru a vedea dacă este guvernată de aceleași modele. După cum vom vedea, în timp ce nu putem trage nicio concluzie din primul scenariu, în cel de-al doilea caz există o tendință vizibilă, similară cu fazele de ajustare-comprimare studiate în IP-uri.

În ceea ce privește hărțile de salienă, în timp ce procesul de superizare acționează în profunzimea rețelei, principiul IB acționează asupra unui singur strat. Pentru a conecta aceste două concepte, verificăm dinamica entropiei spațiale a hărților de salienă pe

parcursul întregului proces de formare, în conjuncție cu comportamentul său pe straturi. Descoperim o anumită formă de continuitate, prezentată în secțiunea următoare.

2.2.3 Experimente și Discuții

În această secțiune, descoperim experimental o legătură între teoria IB de potrivire-compresie și evoluția entropiei spațiale aplicată hărților de salienț bazate pe modele de formare similare. Verificăm, de asemenea, aplicabilitatea practică a modelelor de entropie spațială și o posibilă legătură cu procesul de superizare. Pentru instruire, folosim cadrul de programare a învățării profunde PyTorch [78] (versiunea 1.6.0) și implementarea publică a Grad-CAM, modificată în funcție de nevoile noastre.

Evoluția Entropiei

Derivăm un pic din studiul informației reciproce și analizăm evoluția entropiei spațiale pentru hărțile de salienț în timp pentru aceeași arhitectură VGG16. În timp ce în [73] entropia spațială a fost studiată la un singur moment în timp (după antrenament), mergând de-a lungul adâncimii rețelei, noi ne propunem să surprindem acum dinamica entropiei pe parcursul întregului antrenament, căutând orice tipare.

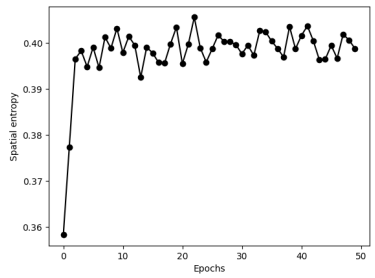
Entropia spațială a fost calculată folosind formulele din secțiunea 1.7 și a fost calculată media pe 50 de eșantioane aleatorii din setul de date de instruire. În tabelul 2.5, există diagrame pentru entropia spațială calculată în timp pentru straturile selectate. Vizualizăm din nou, doar patru straturi.

Acum este vizibil un model, în care entropia spațială crește în timpul fazei inițiale a antrenamentului și la un moment dat se stabilizează. Foarte interesant, am observat aceleași tipare și în cazul altor arhitecturi de rețea bine cunoscute: ResNet [42], DenseNet [48] și GoogleNet [104].

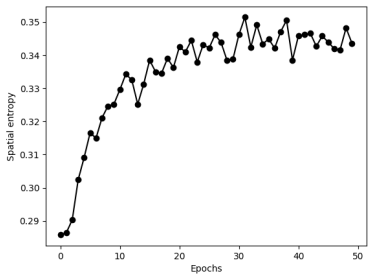
Observăm că straturile timpurii prezintă o creștere mai abruptă a valorilor entropiei în timpul primelor câteva epoci. Acest fenomen ar putea fi atribuit faptului că primele straturi ale unui CNN învață să detecteze concepte ușoare, cum ar fi marginile, iar rețeaua învață să îndeplinească această sarcină mai repede decât ultimele straturi, care sunt responsabile de detectarea unor concepte mai complexe, cum ar fi părți întregi de obiecte [126].

În [5] se afirmă că primele straturi sunt mai rapid de învățat prin utilizarea unei scheme de preînvățare autosupravegheată pe o singură imagine puternic augmentată. Autorii demonstrează că o singură imagine este suficientă pentru a învăța reprezentări bune pentru primele câteva straturi. Împreună cu [5], avem, de asemenea, ipoteza că creșterea bruscă a entropiei, observată pentru primele straturi, se datorează faptului că conceptele mai ușoare sunt învățate mai repede.

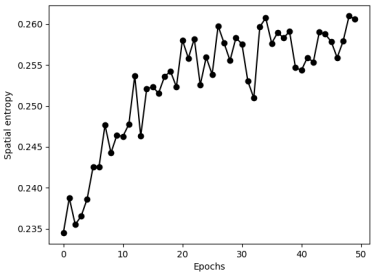
Există totuși câteva excepții de la modelele din tabelul 2.5, prezente doar pentru câteva straturi. Deși nu am reușit încă să găsim o explicație bună pentru aceste modele diferite,



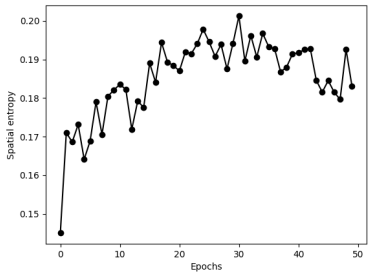
Entropy for 4^{th} layer



Entropy for 12^{th} layer



Entropy for 18^{th} layer



Entropy for 27^{th} layer

Table 2.5: Entropia spațială în timp pentru hărțile de salienț̦ă. Tabel reprodus din [87].

facem următoarea presupunere. La fel ca în cazul [73], în care un întreg strat a fost eliminat dacă nu se producea o scădere a entropiei spațiale, presupunem că am putea elimina un strat dacă entropia spațială nu urmează tiparele prezentate în tabelul 2.5, deoarece acel strat conține informații redundante.

Înghețarea Straturilor în Timpul Antrenamentului

Pe baza modelelor observate în tabelul 2.5, testăm ipoteza că există legături între dinamica entropiei spațiale și evoluția procesului de formare. Ca atare, antrenăm aceeași VGG16 pe setul de date CIFAR-10 și înghețăm straturile în care entropia spațială a hărții de salienț̦ă mediată pe ultimele cinci epoci intră într-o fază de compresie cu variații mici și se situează sub un anumit prag ϵ, și observăm dacă aceasta atinge aceeași acuratețe ca o rețea complet antrenată în același număr de epoci sau într-un număr mai mic de epoci.

Pentru un ϵ mare, straturile sunt înghețate la începutul procesului de instruire, iar învățarea devine prohibitivă. Pentru un ϵ mic, straturile nu sunt, în general, înghețate, iar rețeaua este antrenată ca de obicei. În practică, am constatat că o valoare de ϵ de $5e-05$ funcționează cel mai bine. În tabelul 2.6, sunt prezentate rezultatele empirice pentru o VGG16 complet instruită față de o VGG16 cu unele straturi înghețate în timpul instruirii. Coloana "max accuracy" indică precizia maximă obținută pe setul

de testare CIFAR-10 de către cele două versiuni până la epoca specificată în prima coloană.

Epoca	**Acuratețea maximă**		**Straturi înghețate**	**Timp de rulare (minute)**	
	Înghețat	Normal		Înghețat	Normal
30	85.67%	85.42%	0, 17, 28	66	36
40	86.59%	86.13%	0, 7, 17, 26, 28	88	48
50	86.89%	86.81%	0, 7, 14, 17, 26, 28	110	60
60	87.45%	87.45%	0, 7, 14, 17, 26, 28	133	72

Table 2.6: Performanța VGG16 - straturi normale vs straturi înghețate. Experimentele au fost efectuate pe un GPU Tesla K80 pe Google Colaboratory [9]. Tabel reprodus din [87].

După cum se poate observa, rețeaua este antrenată cu unele straturi înghețate, dar are totuși performanțe la fel de bune sau mai bune decât versiunea cu toate straturile antrenate continuu. Această schemă de instruire poate fi considerată ca o formă de oprire timpurie aplicată la nivel de strat, utilizată de obicei pentru a împiedica o rețea să se adapteze excesiv. Prin urmare, facem o legătură între modelele observate în entropia spațială a hărților de salientă și dinamica de instruire a unei DNN. Observăm că stratul 0, care este primul strat convoluțional, este printre primele care se blochează, ceea ce dovedește empiric ipoteza noastră din subsecțiunea 2.2.3 că primele straturi sunt cele care se învață cel mai repede. Dezavantajul acestei metode este o suprataxare a timpului de execuție a calculului, dar aceasta nu a fost ținta experimentului nostru.

Cel mai asemănător experiment cu al nostru este cel din [119], în care autorii au folosit precizia validării ca indicator pentru a aplica procedura de oprire timpurie și au observat că instruirea poate fi oprită înainte de începerea fazei de compresie. Spre deosebire de lucrarea lor, noi folosim cantități observate direct în dinamica de instruire a rețelei și aplicăm oprirea timpurie pentru a dovedi că aceasta are efect și asupra preciziei de validare.

Crearea unei Legături Între IB și Semiotică

În tabelul 2.7, am observat o proprietate interesantă a entropiei spațiale pentru hărțile de salientă. După ce are loc procesul de superizare (adică după o scădere a valorii entropiei), magnitudinea rezultată după faza de compresie este aproximativ aceeași cu magnitudinea de unde începe entropia înainte de superizare. Am observat o tendință de continuare între straturi, dirijată de evoluția entropiei și de procesul de superizare. Acest lucru ar putea reprezenta o altă proprietate inerentă a dinamicii de instruire a unui DNN: necesitatea de a crește entropia spațială până la o limită superioară determinată de straturile anterioare prin superizare.

Conduși de aceste observații empirice, am observat o legătură interesantă între IB și superizare. Din [73] știm că în interiorul unui DNN are loc un proces de superizare, care

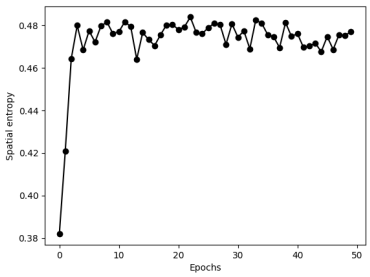
concentrează informația, ceea ce duce la o scădere a entropiei spațiale. Pentru a ajunge la entropia inițială din straturile anterioare, este necesară o creștere a valorii entropiei pentru ultimele straturi. Această creștere poate fi de orice formă: liniară, polinomială, exponențială, dar, după cum se dovedește, urmează foarte îndeaproape aceeași tendință de ajustare și comprimare observată în teoria gâtului de gâtul informației, descrisă în subsecțiunea 2.2.3. Fenomenul vizibil în aceste diagrame este posibil doar dacă există o dependență reciprocă între teoria IB (potrivire-compresie) și superizare. Această observație empirică ar putea explica unele dintre dinamicile de instruire care guvernează DNN-urile moderne, dintr-o perspectivă teoretică a informației.

2.2.4 Concluzii

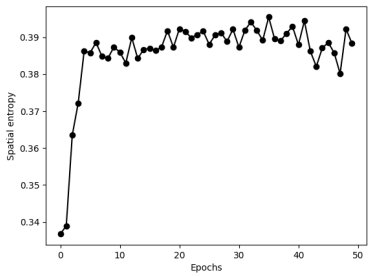
Conform experimentelor noastre, există o legătură între evoluția în timp a entropiei spațiale a hărților de salientă și teoria IB de potrivire-compresie. Am observat o relație de dependență reciprocă între teoria IB și suprapunerea, prezentă în DNN-uri, unde există o scădere a magnitudinii entropiei spațiale și ultimele straturi ajung la aceeași entropie spațială de la care au pornit primele straturi.

Am analizat dacă modelele prezente în entropia spațială afectează dinamica de formare a DNN-urilor. Am observat că unele straturi pot fi oprite mai devreme de la formare, pe baza variabilității entropiei spațiale în timpul fazei de compresie, și încă mai pot obține precizii egale cu cele ale omologilor complet formați. Acest lucru poate fi considerat ca o formă de oprire timpurie, aplicată pe straturi.

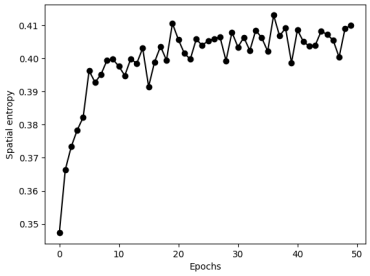
Din câte cunoaștem, aceasta este prima aplicare a conceptului IB la hărțile de salientă și la superizarea semiotică. Experimente suplimentare sunt necesare pentru a trage concluzii mai puternice din observațiile descrise în această lucrare: diferite arhitecturi DNN, aplicații mai practice ale variabilității entropiei spațiale în timp, o înțelegere teoretică mai solidă a fenomenelor descrise în această lucrare.



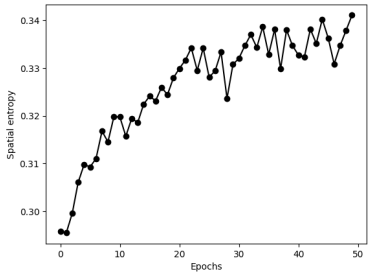
Entropy for 3^{rd} layer



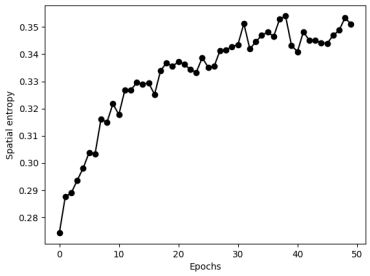
Entropy for 5^{th} layer



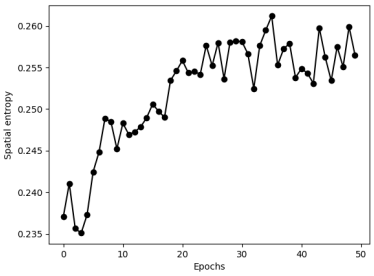
Entropy for 8^{th} layer



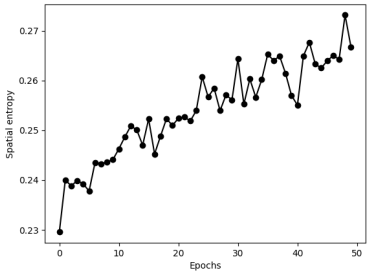
Entropy for 10^{th} layer



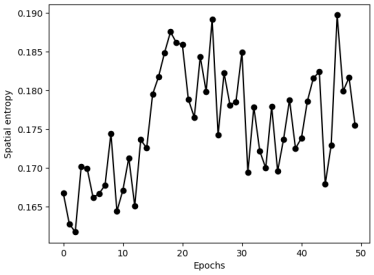
Entropy for 15^{th} layer



Entropy for 17^{th} layer



Entropy for 22^{th} layer



Entropy for 24^{th} layer

Table 2.7: Continuitatea entropiei spațiale pentru hărțile de salienț după suprapunere. Tabel reprodus din [87].

Capitolul 3

Tăierea Rețelelor Neuronale prin Teoretia Informației

3.1 Taierea Filtrelor Convoluționale prin Învățare prin Întărire cu Minimizarea Entropiei

Următoarea secțiune face parte din lucrarea noastră care va fi publicată anul acesta [72]. Textul original reprodus aici face parte din lucrarea noastră și nu este în niciun caz destinat plagiatului sau utilizării fără o atribuire corespunzătoare.

3.1.1 Motivația cercetării

Rețelele neuronale convoluționale moderne (CNN) au apărut odată cu publicarea AlexNet [53] în 2012, care a deschis calea pentru alte arhitecturi precum VGG [97], ResNet [42] și EfficientNet [107]. Deși aceste rețele posedă o capacitate foarte mare și au performanțe la un nivel suprauman, ele sunt adesea supraparametrizate [77], ceea ce induce o latență și un consum de energie ridicat pe dispozitivele alimentate cu baterii. Tehnici precum pruning și cuantificarea [10,20,26,30,117] au devenit recent foarte populare, deoarece pot genera subversiuni eficiente din punct de vedere energetic ale acestor rețele supraparametrizate.

În timp ce pruning-ul se ocupă cu eliminarea greutăților neimportante dintr-o rețea prin aplicarea unei anumite euristici, cuantificarea operează prin utilizarea unui număr mai mic de biți pentru greutăți și activări, accelerând astfel calculele generale. În lucrarea noastră, ne concentrăm doar pe pruningul structurat, care se traduce prin eliminarea unor filtre întregi dintr-un nucleu convoluțional. Pentru aceasta, folosim un cadru de învățare automată a mașinilor (AutoML) pentru a selecta cel mai potrivit procent de sparsity structurat pentru fiecare strat neuronal [45].

AutoML este o strategie puternică utilizată pentru multe sarcini, cum ar fi căutarea arhitecturii neuronale (NAS), căutarea hiperparametrilor, pregătirea datelor, ingineria caracteristicilor [44, 124]. Principiul care stă la baza acesteia este de a automatiza sarcinile de căutare manuală și de a găsi soluții optime mai repede decât putem face manual. Recent, AutoML a fost aplicat pentru comprimarea rețelelor prin pruning [45]. Prin utilizarea unui agent de învățare prin întărire [58], sistemul poate alege automat procentul de raritate pe strat, iar apoi se aplică o euristică de pruning bazată pe magnitudine, care elimină filtrele cu procentajul cel mai ridicat și cu cea mai mică magnitudine. Criteriul de recompensă pe care îl utilizează agentul este acuratețea rețelei obținută pe un subset ales aleatoriu din setul de instruire sau din setul de validare la sfârșitul fazei de tăiere.

Am descoperit că acuratețea rețelei nu este singurul criteriu de recompensă care poate fi utilizat pentru comprimarea rețelei AutoML. Contribuția noastră centrală este o funcție de recompensă teoretică a informației (minimizarea entropiei) pentru agent, care este complet diferită de acuratețea utilizată în [58]. Utilizăm acest criteriu informațional-teoretic pentru curățarea rețelei.

Rezultatul intrigant al activității noastre este descoperirea unei legături interesante între minimizarea entropiei și curățarea structurală. Aceasta ar putea fi legată de măsura entropiei structurale recent introdusă în [2], unde "entropia structurală se referă la nivelul de eterogenitate a nodurilor din rețea, cu premisa că nodurile care împărtășesc funcționalități sau atribute sunt mai conectate decât altele".

În practică, utilizăm cadrul AutoML din [45] pentru a sparsifica o rețea neuronală și propunem ca și criteriu de recompensare a optimizării minimizarea entropiei spațiale (așa cum este definită în [73]) la fiecare strat convoluțional. Prin experimentele noastre, arătăm empiric că această minimizare acționează ca un indicator pentru menținerea acurateței. Noutatea lucrării noastre constă în descoperirea faptului că există și alte abordări, mai principiale, pentru curățarea rețelelor neuronale decât optimizarea directă a acurateței funcției de recompensă a agentului.

3.1.2 O Metodă de Simplificare a Rețelelor Neuronale prin Minimizarea Entropiei Spațiale

Această secțiune descrie metoda noastră, care este o modificare a cadrului AMC [45] pentru curățarea structurală. Principala diferență dintre abordarea noastră și [45] este că funcția de recompensă a agentului nostru minimizează entropia spațială a activărilor convoluționale, în loc să maximizeze acuratețea modelului.

Cadrul AMC este un instrument AutoML pentru pruning care selectează procentul de sparsity pentru fiecare strat al unei rețele neuronale (strat cu strat), apoi un algoritm bazat pe magnitudinea L_2 marchează filtrele cu procentul cel mai mic de magnitudine pentru eliminare. Deoarece acuratețea unui model este o funcție nediferențiabilă pentru care nu se pot calcula și utiliza gradienți în timpul backpropagation, trebuie utilizată o

tehnică RL pentru a optimiza acest criteriu, care va fi tratat ca o funcție de recompensă. Prin urmare, motorul care conduce selecția procentuală pentru tăiere este un agent DDPG [58], antrenat prin intermediul unei tehnici de critică a actorilor [51], folosind precizia calculată pe un set de date separat ca și criteriu de recompensă. Funcția de recompensă poate fi calculată fie pe o parte din setul de date de instruire, fie pe setul de date de validare. Urmărind acțiuni optime (cu recompensă mare), agentul va fi recompensat în mod corespunzător și încurajat să efectueze acțiuni similare în viitor, în timp ce, în general, va descuraja acțiunile cu recompensă slabă.

AMC stochează intrările și ieșirile pentru fiecare strat chiar la începutul optimizării, prin transmiterea unui lot de eșantioane de intrare (numite eșantioane de calibrare). După curățarea cu ajutorul euristicii bazate pe magnitudine, canalele de la intrările eșantioanelor de calibrare care au aceiași indici ca și filtrele eliminate vor fi, de asemenea, eliminate. Se aplică o regresie prin metoda celor mai mici pătrate pentru a ajusta ponderile rămase la noile intrări și la ieșirile deja stocate. După ce cadrul AMC găsește cea mai bună configurație de subrețea care maximizează funcția de recompensă dată și după ce ponderile au fost, de asemenea, ajustate prin intermediul regresiei celor mai mici pătrate, noua configurație a rețelei este ajustată cu precizie, așa cum este standard în literatura de specialitate.

Spre deosebire de formularea AMC originală, noi modificăm recompensa bazată pe precizie prin introducerea unei funcții care minimizează media entropiilor spațiale ale activărilor convoluționale. Scopul nostru este de a observa dacă minimizarea entropiei poate fi utilizată ca un criteriu în locul calculării directe a acurateței, stabilind astfel o legătură potențial interesantă între domeniile pruningului neuronal și teoria informației. Deoarece cadrul AMC încearcă să crească în mod constant cantitatea de recompensă pe care o primește și deoarece formularea entropiei spațiale medii pe care o folosim este delimitată între 0 și 1 [49], o scădem din 1 pentru a minimiza termenul. Astfel, problema de optimizare a agentului devine găsirea gradului de dispersie pentru fiecare strat, ceea ce ar duce în cele din urmă la minimizarea entropiei spațiale. Pentru a calcula valoarea medie (per strat) a entropiei spațiale, utilizăm ieșirile convoluționale din 100 de eșantioane, ceea ce reprezintă, desigur, doar o estimare a întregului set de date. Calcularea entropiei spațiale medii utilizând întregul set de date ar fi prea costisitoare din punct de vedere computațional și, ca atare, este suficient să se recurgă la o dimensiune mai mică a eșantionului.

Ipoteza noastră este că, prin minimizarea entropiei spațiale, putem obține rezultate egale sau mai bune decât atunci când obiectivul este de a maximiza precizia. Dacă acesta este cazul, atunci putem stabili o legătură empirică între pruning și teoria informației, arătând că, prin eliminarea informațiilor redundante dintr-un model, putem obține aceeași precizie ca atunci când încercăm direct să o maximizăm.

3.1.3 Experimente și Discuții

În această secțiune evaluăm experimental conexiunea noastră ipotetică între teoria informației și pruning, verificând dacă pruningul realizat prin AutoML, folosind minimizarea entropiei spațiale pentru activările convoluționale, poate duce la un model mai compact cu o precizie similară.

Pentru instruire, am utilizat cadrul de programare a învățării profunde PyTorch [78] (versiunea 1.10.0) și implementarea publică a AMC, modificată în funcție de nevoile noastre.

Am început prin a antrena un VGG16 standard [97] pe setul de date CIFAR-10 [52]. Pentru aceasta, ne-am antrenat timp de 200 de epoci utilizând optimizatorul SGD cu o rată de învățare de 0.01 și cosine annealing scheduler [64].

Pentru a stabili un punct de referință cu care să comparăm metoda noastră, am utilizat formularea originală a cadrului AMC și am optimizat mai întâi rețeaua utilizând criteriul de precizie. Pentru a atinge un anumit nivel de curățare, AMC împinge în sus nivelul de raritate până când se menține doar un procent predefinit din totalul FLOPS. Raportul dintre numărul de FLOPS după comprimare și numărul de FLOPS înainte de comprimare poate măsura indirect gradul de dispersie dintr-o rețea. Tabelul 3.1 prezintă rezultatele pentru diferite procente de păstrare a FLOPS după reglarea fină pe setul de date CIFAR-10. Pentru ajustarea fină am utilizat același optimizator de formare și aceiași hiperparametri ca și în cazul descris anterior.

	Standard VGG16	VGG16 cu 50% FLOPS	VGG16 cu 20% FLOPS	VGG16 cu 10% FLOPS
Acuratețe	**93.58%**	**93.85%**	**93.26%**	**92.18%**
No. parametrii	**14728266**	**4768242**	**912186**	**483402**

Table 3.1: Precizia pentru o rețea VGG16 care utilizează cadrul AMC original pentru diferite procente de conservare FLOPS. Tabelul original va fi publicat la ICAISC 2023 în una dintre lucrările noastre acceptate.

Observăm că, cu minimizarea entropiei, am obținut aceeași performanță ca atunci când precizia este utilizată ca recompensă. Soluția găsită prin această metodă are cu 10 × mai puțini FLOPS și cu aproximativ 38 × mai puțini parametri decât rețeaua VGG16 originală. Pentru maximizarea entropiei, cadrul produce o soluție care are într-adevăr mai puțini parametri, dar utilizează același număr de FLOPS ca și metoda cu minimizarea entropiei. Putem observa însă că arhitectura de rețea rezultată are o performanță de precizie mult mai slabă.

În experimentele de mai sus, am utilizat o dimensiune de 256 pentru cuantificarea activărilor convoluționale înainte de a calcula entropia spațială.

Pentru a testa generalitatea metodei noastre pentru diverse alte arhitecturi, am repetat

	Minimizare	Maximizare
Acuratețe	**92.36%**	**83.23%**
No. parametrii	**386442**	**91290**

Table 3.2: Experimente cu minimizarea și maximizarea entropiei și conservarea FLOPS a 10%. Preciziile sunt calculate pe setul de testare CIFAR-10 după reglarea fină. Figura originală va fi publicată la ICAISC 2023 într-una dintre lucrările noastre acceptate.

aceleași experimente pentru alte rețele populare: MobileNetV2 [85] și ResNet50 [42]. Rezultatele sunt prezentate în tabelul 3.3. Metoda noastră este la egalitate cu cadrul AMC original pentru diferite arhitecturi și procente de conservare a FLOPS. Singura scădere notabilă a performanțelor este pentru ResNet50, despre care s-a observat anterior că conține mai puțină redundanță [117] și a fost cel mai dificil de comprimat, chiar și atunci când se utilizează precizia ca și criteriu.

Arhitectura	Original performance	Acuratețe 50% FLOPS	Acuratețe 20% FLOPS	Entropie 50% FLOPS	Entropie 20% FLOPS
MobileNetV2	**94.58%**	**94.62%**	**93.59%**	**94.31%**	**93.75%**
ResNet50	**95.21%**	**95.34%**	**95.09%**	**95.27%**	**94.27%**

Table 3.3: Precizia pe setul de teste CIFAR-10 cu alte arhitecturi și diferite procente de conservare a FLOPS. Figura originală va fi publicată la ICAISC 2023 într-una dintre lucrările noastre acceptate.

Utilizând un criteriu de optimizare teoretică a informației, care urmărește minimizarea entropiei, am obținut aceeași performanță ca atunci când optimizăm direct precizia modelului. Am reușit să reducem numărul total de FLOPS al unei arhitecturi VGG16 cu 10 × și numărul de parametri cu aproximativ 38 ×, înregistrând în același timp o scădere minimă a preciziei, cu rezultate similare pentru alte arhitecturi populare.

3.1.4 Concluzii

În mod ideal, rezultatul unei rețele neuronale standard ar trebui să aibă o valoare entropică aproape de zero pentru a prezice cu încredere o clasă - nu neapărat cea corectă. De obicei, acest comportament se obține prin minimizarea entropiei încrucișate dintre ieșirea rețelei și codificarea fierbinte a clasei adevărate. Această sarcină poate fi uneori împovărătoare, deoarece straturile interne ale rețelei nu sunt forțate în niciun fel să minimizeze entropia finală a stratului de ieșire.

În experimentele noastre, am forțat în mod explicit entropia spațială a activărilor convoluționale interne să scadă, cu scopul de a realiza o curățare neuronală. Conform rezultatelor noastre, utilizând entropia spațială ca un criteriu de optimizare într-un cadru de pruning AutoML, putem obține performanțe bune pentru o sarcină de

recunoaștere a obiectelor, fără a optimiza în mod direct metrica de evaluare finală (acuratețea în acest caz). Din cauza supraparametrizării unei rețele neuronale, eliminarea informațiilor neesențiale prin minimizarea entropiei ajută la reducerea modelului la componentele sale relevante (esențiale).

Am stabilit o legătură interesantă între teoria informației și pruningul neuronal. Rezultatul nostru creează premisele pentru viitoarele aplicații în pruningul rețelelor neuronale.

3.2 Accelerarea Tăierii Rețelelor Neuronale Convoluționale prin Intermediul Entropiei Spațiale

Următoarea secțiune se bazează pe lucrarea noastră care va fi publicată în acest an [71]. Textul original reprodus aici face parte din lucrarea noastră și nu este în niciun caz destinat plagiatului sau utilizării fără o atribuire corespunzătoare.

3.2.1 Motivația cercetării

Rețelele neuronale convoluționale profunde au atins performanțe de vârf într-o gamă largă de sarcini de viziune pe calculator, cum ar fi clasificarea imaginilor, detectarea obiectelor și segmentarea semantică [42,53,63]. Aceste modele constau, de obicei, dintr-un număr mare de parametri, ceea ce le face să fie costisitoare din punct de vedere computațional și să necesite multă memorie pentru a fi implementate pe dispozitive cu resurse limitate, cum ar fi telefoanele mobile și sistemele încorporate. Formarea acestor modele necesită resurse și timp de calcul semnificative, ceea ce limitează capacitatea de a explora arhitecturi și hiperparametri la scară largă [101].

O abordare promițătoare pentru a atenua aceste provocări este pruning-ul, care se referă la procesul de reducere a dimensiunii unei rețele neuronale prin eliminarea ponderilor, neuronilor sau filtrelor neimportante, fără pierderi semnificative de precizie [55]. Pruning-ul poate avea ca rezultat modele mai eficiente care necesită mai puțini parametri, consumă mai puțină memorie și au un timp de inferență mai rapid. Acest lucru poate fi deosebit de important pentru aplicațiile în timp real, unde latența și consumul de energie sunt factori critici [122].

În ciuda beneficiilor potențiale ale pruning-ului, există, de asemenea, unele provocări care trebuie abordate. De exemplu, tăierea poate duce la o creștere semnificativă a numărului de iterații de instruire necesare pentru a recupera acuratețea modelului original, ceea ce poate anula beneficiile aduse de dimensiunea redusă a modelului [62]. În plus, alegerea metodei de tăiere, a ratei de tăiere și a strategiei de ajustare fină poate afecta precizia finală și eficiența modelului tăiat [25]. Prin urmare, este esențial să se

proiecteze și să se evalueze cu atenție metodele de pruning pentru diferite aplicații și arhitecturi de rețea.

Sarvani *et al.* a introdus o metodă de curățare a filtrelor bazată pe teoria Information Bottleneck [110] care utilizează Informația reciprocă pentru a determina semnificația filtrelor. Filtrele cu relevanță ridicată (HRel), cele care au un MI mai mare cu etichetele de clasă, sunt considerate mai importante și sunt păstrate. Metoda propusă are performanțe mai bune decât cele mai recente metode de eliminare a filtrelor și a fost demonstrată pe arhitecturile LeNet5, VGG16, ResNet56, ResNet110 și ResNet50, utilizând seturile de date MNIST, CIFAR-10 și ImageNet.

Metoda HRel de estimare a MI se bazează pe estimatorul de entropie alfa al lui Rényi [119]. Cu toate acestea, estimarea MI bazată pe kernel prezintă mai multe probleme, inclusiv selectarea lățimii kernelului, blestemul dimensionalității și complexitatea computațională. Aceste probleme pot duce la o estimare inexactă a MI și limitează aplicarea practică a metodelor bazate pe kernel.

În această lucrare, ne bazăm pe munca lor și propunem o metodă mai eficientă de calcul al MI necesară pentru selecția importanței filtrelor, care reduce timpul de optimizare necesar de la aproape o săptămână de calcul la o singură zi. Metoda se bazează pe formularea MI, așa cum a fost definită în lucrarea noastră anterioară [87], bazată pe entropia aurei spațiale. Metoda de entropie a aurei spațiale este mai eficientă și mai simplă decât estimatorii pe bază de kernel și nu necesită selectarea lățimii kernelului. Metoda noastră reușește să păstreze sau să îmbunătățească rezultatele obținute de lucrarea originală, dar la un cost de calcul mult mai mic.

3.2.2 O Metodă de Accelerare a Estimării Informației Reciproce Folosind Entropia Spațială

În această secțiune prezentăm metoda noastră de tăiere a filtrelor CNN. În primul rând, aceasta urmează fluxul de lucru HRel [86].

Entropia lui Y este garantată a fi 0 deoarece Y este codificat utilizând codificarea cu un singur foc, care plasează toată magnitudinea pe o singură poziție. Ca atare, în cazul nostru, formula MI devine:

La fel ca în [86], calculăm MI între fiecare hartă de activare și adevărurile de bază codificate printr-un singur foc. MI între două variabile aleatoare X și Y se calculează astfel:

$$I(X,Y) = H(X) + H(Y) - H(X,Y) \tag{3.1}$$

$$I(X,Y) = H(X) - H(X,Y) \tag{3.2}$$

Pentru a calcula valoarea ecuației (3.2), utilizăm rezultatul ecuației (1.6) pentru entropia lui X, precum și ecuația (1.5) pentru entropia comună a lui X și Y.

În primul rând, antrenăm un CNN până la convergență. Apoi, calculăm MI între hărțile de activare și etichetele de bază reale pentru fiecare strat al rețelei. Hărțile de activare reprezintă ieșirea fiecărui filtru din strat atunci când intrarea este trecută prin acest strat.

În continuare, clasificăm filtrele în ordinea descrescătoare a MI și eliminăm un anumit procent de filtre cu cel mai mic MI. Numărul de filtre care urmează să fie eliminate este determinat pentru fiecare strat pe baza obiectivului total maxim de filtre eliminate și a contribuției stratului la performanța generală a rețelei.

După eliminarea filtrelor, efectuăm o etapă de reglare fină pentru a recupera performanța inițială a rețelei. Reglarea fină presupune antrenarea rețelei curățate pentru câteva epoci folosind o rată de învățare mai mică decât rata de învățare inițială.

După finalizarea etapei de reglare fină, repetăm procesul de calculare a MI, a filtrelor de clasificare și a MI. Acest proces continuă până când se atinge numărul maxim de filtre eliminate pentru fiecare strat.

Avantajul utilizării MI ca și criteriu de tăiere este acela că ia în considerare conținutul informațional al fiecărui filtru din rețea, mai degrabă decât doar mărimea ponderii sau gradientul acestuia. În plus, MI poate surprinde interacțiunile dintre filtre și contribuția acestora la performanța globală a rețelei.

În metoda HRel, MI este estimată folosind estimatorul alfa entropiei lui Rényi [119], care se bazează pe funcții de kernel și este foarte sensibilă la lățimea de bandă optimă a kernelului din setul de date [109], după cum au observat chiar autorii. Această sensibilitate are ca rezultat o procedură costisitoare din punct de vedere computațional, deoarece lățimea de bandă a nucleului trebuie actualizată în mod continuu în timpul instruirii și al reglajului fin, ceea ce face ca mecanismul de tăiere să fie împovărător. Ca atare, tăierea unei rețele folosind codul public furnizat de autori durează aproape o săptămână pentru o rețea VGG16 [97], ceea ce o face impracticabilă pentru aplicațiile din viața reală.

Spre deosebire de tehnica originală HRel, metoda noastră utilizează o abordare diferită prin utilizarea unei estimări MI care nu se bazează pe funcții kernel. În schimb, estimăm MI folosind entropia spațială a aurei (versiunea simplificată AME) descrisă în secțiunea 1.7. Acest lucru permite calcularea eficientă a MI folosind doar 100 de eșantioane. Eliminăm sarcina de calcul a actualizării continue a lățimii de bandă a nucleului în timpul antrenamentului și al reglajului fin, ceea ce îl face semnificativ mai rapid și mai practic pentru aplicațiile din viața reală. Metoda noastră păstrează sau chiar îmbunătățește rezultatele obținute în lucrarea originală a HRel, demonstrând eficacitatea sa în selectarea eficientă a filtrelor.

3.2.3 Experimente și Discuții

Această secțiune prezintă dovada empirică a eficacității abordării noastre. Evaluăm metoda noastră pe setul de date de referință CIFAR-10, utilizat pe scară largă [52]. Am modificat codul disponibil public furnizat de autorii HRel [86], modificând procedura de estimare a IM. Deși am observat unele variații în performanța de bază a arhitecturilor ResNet față de cea raportată în lucrarea originală, am comparat metoda noastră și HRel folosind aceeași performanță de bază inițială.

VGG16

Pentru a evalua eficacitatea metodei noastre propuse, o aplicăm la arhitectura populară VGG16 [97], care constă din 13 straturi convoluționale și două straturi complet conectate. Eliminăm filtrele din straturile convoluționale și antrenăm rețeaua timp de 300 de epoci cu o rată de învățare inițială de 0,1, pe care o reducem cu un factor de 10 la numerele de epoci 80, 140 și 230, până când se obține precizia de bază. Ulterior, curățăm și antrenăm din nou rețeaua pentru 90 de epoci cu o rată de învățare de 0,01, pe care o reducem cu un factor de 10 la epocile 40 și 70.

Tabelul 3.4 prezintă o comparație între precizia obținută de formularea HRel originală și metoda noastră propusă pentru diferite configurații ale numărului rămas de filtre pe strat pe setul de testare CIFAR-10. Metoda noastră este mai performantă decât HRel pentru ambele configurații, demonstrând eficacitatea sa în procesul de tăiere. Mai exact, rețeaua VGG16 originală atinge o precizie de testare de 93,95%, în timp ce metoda noastră propusă obține precizii de testare de 93,22% și 93,4% pentru configurația 1 și, respectiv, configurația 2. Aceste rezultate evidențiază potențialul metodei noastre de îmbunătățire a eficienței și acurateței modelului.

	HRel	**Metoda noastră**
Rețeaua originală: 64-64-128-128-256-256-256 -512-512-512-512-512-512	93.95%	93.95%
Configurația 1: 19-48-64-64-95-107-107 -175-71-71-44-44-56	93.15%	93.22%
Configurația 2: 24-40-64-77-176-134-120 -141-56-56-56-56-56	93.22%	93.4%

Table 3.4: Compararea acurateței obținute de HRel și de metoda noastră pe setul de testare CIFAR-10 în funcție de diferite configurații de tăiere. Tabelul arată că metoda noastră este mai performantă decât HRel pentru ambele configurații, demonstrând eficacitatea sa în procesul de curățare. Figura originală va fi publicată la IV2023 în una dintre lucrările noastre acceptate.

ResNet56

ResNet56 [42] este o arhitectură de rețea neuronală mai complexă și mai profundă în comparație cu VGG16. Aceasta constă din 55 de straturi convoluționale și 1 strat complet conectat, toate straturile convoluționale (cu excepția primului) fiind grupate în trei blocuri, fiecare conținând 18 straturi convoluționale. Primul, al doilea și al treilea bloc au 16, 32 și, respectiv, 64 de filtre. Pentru a obține acuratețea de referință, rețeaua este antrenată timp de 180 de epoci cu o rată de învățare inițială de 0,1, care este apoi redusă cu un factor de 10 la numerele de epoci 91 și 136.

După curățare, rețeaua este antrenată din nou pentru 200 de epoci cu o rată de învățare de 0,01, care este apoi redusă cu un factor de 10 la epocile 100 și 150. Observăm că arhitectura ResNet56 necesită un număr dublu de epoci de reglaj fin folosind metoda noastră, în comparație cu cadrul HRel original. Cu toate acestea, timpul de calcul suplimentar este neglijabil în comparație cu timpul total de execuție al procesului de curățare din cadrul HRel original.

După tăiere, numărul final de filtre rămase în straturile convoluționale ale fiecărui bloc este de 8, 15 și, respectiv, 30. Tabelul 3.5 arată că metoda noastră este mai performantă decât HRel în ceea ce privește numărul de filtre rămase și a obținut o precizie similară. Mai exact, cadrul original HRel atinge o precizie de 92,74% cu o configurație de filtre de 8-16-32, în timp ce metoda noastră atinge o precizie de 92,76% cu o configurație de filtre de 8-15-30.

	HRel	**Metoda noastră**
Rețeaua originală: 16-32-64	93.45%	93.45%
Configurația: 8-15-30	92.74%	92.76%

Table 3.5: Comparația preciziei obținute de HRel și de metoda noastră pe setul de testare CIFAR-10 utilizând arhitectura ResNet56 cu configurația de filtrare 8-15-30 după tăiere. Figura originală va fi publicată la IV2023 în una dintre lucrările noastre acceptate.

ResNet110

ResNet110 [42] este o arhitectură de rețea neuronală profundă care este compusă din 109 straturi convoluționale și un singur strat complet conectat. Structura ResNet110 este similară cu cea a ResNet56, în care straturile convoluționale sunt grupate în trei blocuri, cu excepția primului strat convoluțional. Cu toate acestea, în ResNet110, fiecare bloc conține 36 de straturi convoluționale cu 16, 32 și, respectiv, 64 de filtre.

Pentru a obține acuratețea de referință, rețeaua este antrenată timp de 240 de epoci cu o rată de învățare inițială de 0,1, care este redusă cu un factor de 10 la numerele de

epoci 88, 160 și 190. Primul strat convoluțional nu este curățat, similar cu alte metode de curățare. După tăiere, rețeaua este ajustată fin pentru 70 de epoci cu o rată de învățare de 0,01, care este redusă cu un factor de 10 la epocile 30 și 50. Numărul de filtre rămase în stratul convoluțional al fiecărui bloc este de 8, 15 și, respectiv, 30, după tăiere.

Tabelul 3.6 prezintă o comparație a preciziei obținute de metoda noastră propusă și de HRel pe setul de testare CIFAR-10 folosind arhitectura ResNet110 cu o configurație a filtrelor de 8-15-30 după tăiere. Rețeaua originală cu o configurație de filtrare de 16-32-64 a obținut o precizie de 93,27%. Metoda noastră a obținut o precizie ridicată de 92,42%, care este ușor îmbunătățită în comparație cu precizia HRel de 92,36%, cu o diferență de 0,06%.

	HRel	**Metoda noastră**
Rețeaua originală: 16-32-64	93.27%	93.27%
Configurația: 8-15-30	92.36%	92.42%

Table 3.6: Comparație între acuratețea obținută de HRel și metoda propusă de noi pe setul de testare CIFAR-10 utilizând arhitectura ResNet110 cu o configurație de filtre de 8-15-30 după tăiere. Figura originală va fi publicată la IV2023 în una dintre lucrările noastre acceptate.

Experimentele demonstrează eficacitatea metodei noastre în îmbunătățirea acurateței și eficienței de tăiere a CNN. Încorporarea entropiei aurei spațiale în calculul MI oferă o metodă de tăiere mai robustă și mai eficientă. Acest lucru este realizat prin îmbunătățirea acurateței criteriilor de selecție a importanței filtrelor și prin reducerea timpului de optimizare necesar pentru calculul MI. Metoda noastră nu numai că este mai performantă decât metoda HRel de bază în ceea ce privește performanța de tăiere, dar reduce semnificativ și costul de calcul, ceea ce o face o soluție practică și scalabilă pentru comprimarea modelelor de învățare profundă.

3.2.4 Concluzii

Am introdus o soluție alternativă la estimatorul de entropie alfa al lui Rényi bazat pe matrice, utilizat în metoda HRel propusă în [86]. Această îmbunătățire reduce semnificativ timpul de optimizare de la aproape o săptămână la o singură zi, ceea ce o face o metodă mai practică și mai eficientă pentru curățarea modelelor la scară largă. Metoda noastră este o soluție eficientă și eficace pentru reducerea complexității computaționale și a amprentei de memorie a modelelor de învățare profundă, oferind o alternativă viabilă la metodele existente cu performanțe îmbunătățite de pruning și eficiență computațională.

Capitolul 4

Concluzie

4.1 Concluzie

Lucrările prezentate în această teză au un fir comun care se învârte în jurul subiectului teoriei informației, în special cu accent pe entropie. Primele două articole din această colecție explorează intersecția dintre semiotică și învățarea profundă prin examinarea modelelor fluctuante ale entropiei spațiale în cadrul hărților de salienţă. Pe de altă parte, ultimele două articole propun abordări practice care încorporează entropia spațială în domeniul pruningului rețelelor neuronale. În consecință, pentru fiecare articol în parte, prezentăm o prezentare generală cuprinzătoare a principalelor contribuții și oferim concluzii relevante.

Rezultatele prezentate în secțiunea 2.1 sunt centrate în jurul unei aplicații noi a semioticii computaționale în analiza și interpretarea rețelelor neuronale profunde, care aduce o perspectivă nouă în domeniu. Prin integrarea conceptelor din semiotică, o disciplină axată pe semne și utilizarea lor, oferim o înțelegere unică a modului în care se desfășoară procesele decizionale în modelele CNN. În plus, explorăm potențialul de a valorifica instrumentele semiotice pentru a optimiza arhitectura rețelelor neuronale de învățare profundă, un domeniu care face în prezent obiectul unor investigații active în domeniul învățării automate.

Secțiunea 2.2 prezintă investigația noastră, în care am găsit o legătură convingătoare între evoluția entropiei spațiale în hărțile de salienţă și teoria IB de potrivire-comprimare. Prin intermediul experimentelor noastre, am observat o relație de dependență reciprocă între teoria IB și superizarea semiotică în rețelele neuronale profunde. Mai exact, pe măsură ce am progresat prin straturile rețelei, am observat o scădere semnificativă a magnitudinii entropiei spațiale, straturile ulterioare ajungând la același nivel de entropie spațială ca și straturile anterioare. Această constatare sugerează că principiul IB de comprimare a informațiilor relevante joacă un rol în procesul de superizare a rețelelor DNN.

În secțiunea 3.1, contribuția noastră centrală este introducerea unei funcții de recompensă teoretică a informației bazată pe minimizarea entropiei pentru comprimarea rețelei AutoML. Acest criteriu nou diferă de abordarea tradițională axată pe acuratețe și oferă o alternativă de principiu pentru tăierea rețelelor. Prin luarea în considerare a entropiei activărilor neuronale din straturile ascunse, exploatăm o sursă bogată de informații care este adesea neglijată în contextul tăierii. În timp ce entropia încrucișată este utilizată în mod obișnuit pentru a măsura eroarea dintre rezultatul prezis al unei rețele și distribuția reală a clasei, noi susținem că entropia activărilor neuronale oferă informații valoroase despre distribuția informațiilor în cadrul rețelei.

În cele din urmă, în secțiunea 3.2, prezentăm o metodă nouă și eficientă de calculare a informației reciproce pentru selectarea importanței filtrelor în contextul reducerii modelelor. Pornind de la metoda de tăiere a filtrelor bazată pe teoria Information Bottleneck propusă de Sarvani și colab., abordarea noastră abordează limitările estimatorilor de entropie alfa Rényi's alfa existenți pe bază de matrice. Prin valorificarea conceptului de entropie de aură spațială din lucrările noastre anterioare, elaborăm o soluție mai practică și mai eficientă din punct de vedere computațional pentru curățarea modelelor la scară largă. În mod remarcabil, metoda noastră reduce în mod semnificativ timpul de optimizare de la aproape o săptămână la o singură zi, menținând sau depășind în același timp performanțele de curățare obținute de abordările anterioare.

4.2 Lucrări Viitoare

Rezultatele prezentate în secțiunea 2.1 reprezintă o aplicație originală a semioticii computaționale în analiza și interpretarea rețelelor neuronale profunde, ceea ce, după cunoștințele noastre, nu a mai fost făcut până acum. În lucrările viitoare, ar fi benefică extinderea analizei noastre dincolo de concentrarea exclusivă asupra rețelelor neuronale convoluționale (CNN), care sunt utilizate în principal pentru procesarea imaginilor. Deși studiul nostru actual s-a concentrat pe CNN-uri datorită relevanței lor, este important să explorăm conexiunile cu alte domenii, cum ar fi audio și text, precum și să investigăm diferite tipuri de arhitectură, cum ar fi rețelele neuronale recurente. În plus, abordarea semiotică pe care am utilizat-o poate fi extinsă la diferite modele de învățare profundă, deoarece conceptul de suprapunere semiotică pare să fie prezent în multe arhitecturi. În general, cadrul semioticii computaționale se arată promițător în ceea ce privește contribuția la explicarea și optimizarea rețelelor profunde, în special în cazurile în care sunt implicate mai multe niveluri de superizare.

Contribuțiile prezentate în secțiunea 2.2 pot fi împărțite în două părți. În primul rând, se stabilește o legătură între ipoteza IB de potrivire și compresie și superizarea semiotică prin examinarea evoluției entropiei spațiale în hărțile de salienă. În al doilea rând, se concepe o strategie euristică de formare pentru oprirea timpurie a straturilor pe baza variabilității entropiei spațiale în timp. Această strategie poate fi utilizată în mod practic pentru a preveni supraajustarea în timpul procesului de învățare. Pentru a

spori validitatea rezultatelor noastre, sunt necesare experimente suplimentare. Aceste experimente viitoare ar trebui să implice un spectru mai larg de arhitecturi DNN, să exploreze mai în profunzime aplicațiile practice ale variabilității entropiei spațiale în timp și să contribuie la o înțelegere teoretică mai cuprinzătoare a fenomenelor investigate în acest studiu.

Contribuția originală prezentată în secțiunea 3.1 presupune introducerea unei abordări noi a pruningului rețelelor neuronale într-un cadru AutoML existent. În acest context, este propus un criteriu de recompensă de optimizare, centrat pe minimizarea entropiei spațiale la fiecare strat convoluțional. Experimente empirice extinse stabilesc eficacitatea acestei strategii de minimizare a entropiei în menținerea nivelurilor de acuratețe. Semnificația acestei lucrări constă în explorarea unor metode alternative și mai principiale de curățare a rețelelor neuronale, care se îndepărtează de abordarea tradițională de optimizare directă a funcției de recompensă a agentului pe baza acurateței. În cercetările viitoare, ne vom concentra pe explorarea unor abordări inovatoare pentru utilizarea entropiei în optimizarea arhitecturilor neuronale. O cale potențială este luarea în considerare a măsurii entropiei ca euristică pentru selectarea canalelor conservate, în loc să se bazeze exclusiv pe criteriul de magnitudine L2 utilizat în mod obișnuit. În plus, ne propunem să investigăm dacă există o legătură între tunderea bazată pe entropie și agregarea semiotică. Aceste direcții reprezintă domenii promițătoare de aprofundare și pot oferi informații suplimentare în ceea ce privește optimizarea și înțelegerea rețelelor neuronale.

Lucrările prezentate în secțiunea 3.2 contribuie la abordarea limitărilor metodelor de estimare a informației reciproce bazate pe kernel, inclusiv a problemelor legate de selectarea lățimii kernelului, a blestemului dimensionalității și a complexității computaționale. Pornind de la metoda HRel, care utilizează estimatorul alfa entropiei lui R [86], se propune o abordare mai eficientă pentru calcularea MI, în special pentru selectarea importanței filtrelor. Metoda propusă reduce semnificativ timpul de optimizare necesar, de la aproape o săptămână de calcul la o singură zi. Bazându-se pe formularea MI definită în lucrarea anterioară [87], care utilizează entropia aurei spațiale, metoda propusă oferă o alternativă mai eficientă și mai directă la estimatorii pe bază de kernel, fără a fi nevoie de selectarea lățimii kernelului. În mod notabil, rezultatele obținute prin metoda propusă le mențin sau chiar le îmbunătățesc pe cele obținute prin lucrarea originală, reducând în același timp costurile de calcul într-o mare măsură. Lucrările viitoare s-ar putea concentra pe explorarea aplicabilității metodei noastre pe alte seturi de date de referință și arhitecturi de model, precum și pe investigarea potențialului său în alte domenii ale învățării automate și ale vederii pe calculator.

Bibliography

[1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems*, pages 9505–9515, 2018.

[2] Assaf Almog and Erez Shmueli. Structural entropy: monitoring correlation-based networks over time with application to financial markets. *Scientific reports*, 9(1):1–13, 2019.

[3] Răzvan Andonie. A semiotic approach to hierarchical computer vision. In J. Ross, editor, *Cybernetics and Systems (Proceedings of the Seventh International Congress of Cybernetics and Systems, London, Sept. 7-11, 1987)*, pages 930–933, Lytham St. Annes, U.K., 1987. Thales Publication.

[4] Răzvan Andonie. Semiotic aggregation in computer vision. *Revue roumaine de linguistique, Cahiers de linguistique théorique et appliquée*, 24:103–107, 1987.

[5] YM Asano, C Rupprecht, and A Vedaldi. A critical analysis of self-supervision, or what we can learn from a single image. In *International Conference on Learning Representations*, 2020.

[6] Roland Barthes. *Image, Music, Text*. Hill and Wang, 1977.

[7] David Bau, Bolei Zhu, Hendrik Strobelt, Bo Zhou, Joshua B Tenenbaum, William T Freeman, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6541–6549, 2017.

[8] Max Bense. *Semiotische Prozesse und Systeme in Wissenschaftstheorie und Design, Ästhetik und Mathematik*. Agis-Verlag, Baden-Baden, 1975.

[9] Ekaba Bisong. *Google Colaboratory*. Apress, Berkeley, CA, 2019.

[10] Davis Blalock, Jose Javier Gonzalez Ortiz, Jonathan Frankle, and John Guttag. What is the state of neural network pruning?, 2020.

[11] Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, ..., and Xi Zhang. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016.

[12] Norman Bryson. *Vision and Painting: The Logic of the Gaze*. Yale University Press, 1994.

[13] Daniel Chandler. *Semiotics: The basics*. Taylor & Francis, 2017.

[14] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):834–848, 2017.

[15] Xiaotong Chen, Li Liu, Jing Yang, Chang Yang, and Wangmeng Zuo. Semantics-assisted interpretation of deep neural networks for visual recognition. *Pattern Recognition*, 104:107312, 2020.

[16] Yu Cheng, Chen Gao, Guoqiang Xu, and Xin Lu. Exploration of information in deep neural networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1–9, 2015.

[17] Yu Cheng, Duo Wang, Pan Zhou, and Tao Zhang. A survey of model compression and acceleration for deep neural networks. *CoRR*, abs/1710.09282, 2017.

[18] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, 2012.

[19] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[20] Lei Deng, Guoqi Li, Song Han, Luping Shi, and Yuan Xie. Model compression and hardware acceleration for neural networks: A comprehensive survey. *Proceedings of the IEEE*, 108(4):485–532, 2020.

[21] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186, 2018.

[22] Umberto Eco. *A Theory of Semiotics*. Indiana University Press, 1976.

[23] Jiashi Feng, Junzhou Huang, Ivan Laptev, and Shuicheng Wang. Semantic adversarial deep learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11649–11658, 2020.

[24] Helmar Frank. *Kybernetische Grundlagen der Pädagogik: eine Einführung in die Informationspsychologie und ihre philosophischen, mathematischen und physiologischen Grundlagen*. Agis-Verlag, Baden-Baden, second edition, 1969.

[25] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*, 2019.

[26] Trevor Gale, Erich Elsen, and Sara Hooker. The state of sparsity in deep neural networks, 2019.

[27] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3354–3361. IEEE, 2012.

[28] Bernhard C. Geiger. On information plane analyses of neural network classifiers - a review. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–13, 2021.

[29] Robert Geirhos, Dann Janssen, Heiko H Schütt, Jonas Rauber, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(10):665–673, 2020.

[30] Amir Gholami, Sehoon Kim, Zhen Dong, Zhewei Yao, Michael W. Mahoney, and Kurt Keutzer. A survey of quantization methods for efficient neural network inference. *ArXiv*, abs/2103.13630, 2022.

[31] Yoav Goldberg and Graeme Hirst. *Neural Network Methods in Natural Language Processing*. Morgan & Claypool Publishers, 2017.

[32] Ziv Goldfeld and Yury Polyanskiy. The information bottleneck problem and its applications in machine learning. *CoRR*, abs/2004.14941, 2020.

[33] Antônio Gomes, Ricardo Gudwin, and João Queiroz. Towards meaning processes in computers from peircean semiotics. *SEED Journal–Semiotics, Evolution, Energy, and Development*, 3(2):69–79, 2003.

[34] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[35] Algirdas Julien Greimas. *Structural Semantics: An Attempt at a Method*. University of Nebraska Press, 1983.

[36] Ricardo Gudwin and Fernando Gomide. A computational semiotics approach for soft computing. In *1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*, volume 4, pages 3981–3986. IEEE, 1997.

[37] Ricardo Gudwin and João Queiroz. Towards an introduction to computational semiotics. In *International Conference on Integration of Knowledge Intensive Multi-Agent Systems*, pages 393–398. IEEE, 2005.

[38] Ricardo R Gudwin. Semiotic synthesis and semionic networks. *SEED Journal (Semiotics, Evolution, Energy, and Development)*, 2:55–83, 2002.

[39] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A survey of methods for explaining black box models. *ACM Computing Surveys (CSUR)*, 51(5):93, 2018.

[40] Song Han, Jeff Pool, John Tran, and William J Dally. Learning both weights and connections for efficient neural networks. In *Advances in neural information processing systems*, pages 1135–1143, 2015.

[41] B. Hassibi, D.G. Stork, and G.J. Wolff. Optimal brain surgeon and general network pruning. In *IEEE International Conference on Neural Networks*, pages 293–299 vol.1, 1993.

[42] K He, X Zhang, S Ren, and J Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[43] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 346–361, Cham, 2014. Springer International Publishing.

[44] Xin He, Kaiyong Zhao, and Xiaowen Chu. Automl: A survey of the state-of-the-art. *Knowledge-Based Systems*, 212:106622, 2021.

[45] Yihui He, Ji Lin, Zhijian Liu, Hanrui Wang, Li-Jia Li, and Song Han. Amc: Automl for model compression and acceleration on mobile devices. In *European Conference on Computer Vision (ECCV)*, 2018.

[46] Yihui He, Xiangyu Zhang, and Shaoqing Sun. Channel pruning for accelerating very deep neural networks. *International Conference on Computer Vision*, pages 1398–1406, 2017.

[47] John R Hershey, Zhuo Chen, Jonathan Le Roux, and Shinji Watanabe. Deep clustering: Discriminative embeddings for segmentation and separation. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 31–35. IEEE, 2016.

[48] Gao Huang, Zhuang Liu, and Kilian Q. Weinberger. Densely connected convolutional networks. *CoRR*, abs/1608.06993, 2016.

[49] A. G. Journel and C. V. Deutsch. Entropy and spatial disorder. *Mathematical Geology*, 25(3):329–355, 1993.

[50] I. Kokkinos. Ubernet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5454–5463, 2017.

[51] Vijay Konda and John Tsitsiklis. Actor-critic algorithms. In S. Solla, T. Leen, and K. Müller, editors, *Advances in Neural Information Processing Systems*, volume 12. MIT Press, 1999.

[52] Alex Krizhevsky. Learning multiple layers of features from tiny images. University of Toronto, 2012.

[53] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[54] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998.

[55] Yann LeCun, John Denker, and Sara Solla. Optimal brain damage. In D. Touretzky, editor, *Advances in Neural Information Processing Systems*, volume 2. Morgan-Kaufmann, 1989.

[56] Hao Li, Asit Kadav, Ivana Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. In *Proceedings of the International Conference on Learning Representations*, 2017.

[57] Li Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 Conference on Computer Vision and Pattern Recognition Workshop*, pages 178–178, 2004.

[58] Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. In Yoshua Bengio and Yann LeCun, editors, *ICLR*, 2016.

[59] Henry W Lin and Max Tegmark. Towards understanding the role of over-parametrization in generalization of neural networks. *arXiv preprint arXiv:1705.11029*, 2017.

[60] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. Ssd: Single shot multibox detector. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 21–37, Cham, 2016. Springer International Publishing.

[61] Zhuang Liu, Jianguo Li, Zhiqiang Shen, Gao Huang, Shoumian Yan, and Changshui Zhang. Learning efficient convolutional networks through network slimming. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2736–2744, 2017.

[62] Zhuang Liu, Mingjie Sun, Tinghui Zhou, Gao Huang, and Trevor Darrell. Rethinking the value of network pruning. In *International Conference on Learning Representations*, 2019.

[63] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[64] Ilya Loshchilov and Frank Hutter. SGDR: stochastic gradient descent with warm restarts. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[65] Ali Mahdi, Jing Qin, and George Crosby. DeepFeat: A bottom-up and top-down saliency model based on deep features of convolutional neural networks. *IEEE Transactions on Cognitive and Developmental Systems*, 12(1):54–63, 2020.

[66] Volodymyr Mnih, Adrià Puigdomènech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. *arXiv preprint arXiv:1602.01783*, 2016.

[67] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

[68] Decebal Constantin Mocanu, Elena Mocanu, Peter Stone, Phuong H Nguyen, Madeleine Gibescu, and Antonio Liotta. Scalable training of artificial neural networks with adaptive sparse connectivity inspired by network science. *Nature Communications*, 9(1):1–11, 2018.

[69] C.W. Morris and M. Charles William. *Writings on the General Theory of Signs*. Approaches to semiotics. Mouton, 1972.

[70] Michael C Mozer and Paul Smolensky. Skeletonization: A technique for trimming the fat from a network via relevance assessment. In D. Touretzky, editor, *Advances in Neural Information Processing Systems*, volume 1. Morgan-Kaufmann, 1988.

[71] Bogdan Musat and Razvan Andonie. Accelerating convolutional neural network pruning via spatial aura entropy. In *To be published in the proceedings of the 27 International Conference Information Visualisation*, 2023.

[72] Bogdan Musat and Razvan Andonie. Pruning convolutional filters via reinforcement learning with entropy minimization. In *To be published in the proceedings of the 22nd International Conference on Artificial Intelligence and Soft Computing*, 2023.

[73] Bogdan Muşat and Răzvan Andonie. Semiotic aggregation in deep learning. *Entropy*, 22(12), 2020.

[74] Hai Nguyen and Hung La. Review of deep reinforcement learning for robot manipulation. In *2019 Third IEEE International Conference on Robotic Computing (IRC)*, pages 590–595, 2019.

[75] Niall O' Mahony, Sean Campbell, Anderson Carvalho, Suman Harapanahalli, Gustavo Adolfo Velasco-Hernández, Lenka Krpalkova, Daniel Riordan, and Joseph Walsh. Deep learning vs. traditional computer vision. *CoRR*, abs/1910.13796, 2019.

[76] Samet Oymak and Mahdi Soltanolkotabi. Towards moderate overparameterization: global convergence guarantees for training shallow neural networks. *CoRR*, abs/1902.04674, 2019.

[77] Samet Oymak and Mahdi Soltanolkotabi. Toward moderate overparameterization: Global convergence guarantees for training shallow neural networks. *IEEE Journal on Selected Areas in Information Theory*, 1:84–105, 2020.

[78] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8026–8037, 2019.

[79] Charles S. Peirce. *Collected papers of Charles Sanders Peirce*, volume 2. Harvard University Press, 1960.

[80] Harry A. Pierson and Michael S. Gashler. Deep learning in robotics: A review of recent research. 2017.

[81] Q. R. Razlighi, N. Kehtarnavaz, and A. Nosratinia. Computation of image spatial entropy using Quadrilateral Markov Random Field. *IEEE Transactions on Image Processing*, 18:2629–2639, 2009.

[82] Qolamreza Razlighi and Nasser Kehtarnavaz. Fast computation methods for estimation of image spatial entropy. *J. Real-Time Image Processing*, 6:137–142, 06 2011.

[83] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, page 91–99, Cambridge, MA, USA, 2015. MIT Press.

[84] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.

[85] Mark Sandler, Andrew G. Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. *CoRR*, abs/1801.04381, 2018.

[86] C.H. Sarvani, Mrinmoy Ghorai, Shiv Ram Dubey, and S.H. Shabbeer Basha. Hrel: Filter pruning based on high relevance between activation maps and class labels. *Neural Networks*, 147:186–197, 2022.

[87] Bogdan Mușat and Răzvan Andonie. Information bottleneck in deep learning - a semiotic approach. *International Journal of Computers Communications & Control*, 17(1), 2022.

[88] Ferdinand de Saussure. *Course in General Linguistics*. Philosophical Library, New York, NY, 1916.

[89] Andrew Michael Saxe, Yamini Bansal, Joel Dapello, Madhu Advani, Artemy Kolchinsky, Brendan Daniel Tracey, and David Daniel Cox. On the information bottleneck theory of deep learning. In *International Conference on Learning Representations (ICLR)*, 2018.

[90] T.A. Sebeok. *Signs: An Introduction to Semiotics*. Toronto Studies in Semiotics. University of Toronto Press, 1994.

[91] R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, and D. Batra. Grad-CAM: Why did you say that? Visual explanations from deep networks via gradient-based localization. In *CoRR*, volume abs/1610.02391, 2016.

[92] Dinggang Shen, Guorong Wu, and Heung-Il Suk. Deep learning in medical image analysis. *Annual Review of Biomedical Engineering*, 19:221–248, 2017.

[93] Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information. *CoRR*, abs/1703.00810, 2017.

[94] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

[95] David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. Deterministic policy gradient algorithms. *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 387–395, 2014.

[96] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *CoRR*, abs/1312.6034, 2013.

[97] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

[98] Dominic Smilkov, Nikhil Thorat, Been Kim, Fernanda B. Viégas, and Martin Wattenberg. SmoothGrad: removing noise by adding noise. *CoRR*, abs/1706.03825, 2017.

[99] J.T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller. Striving for simplicity: The all convolutional net. In *ICLR (workshop track)*, 2015.

[100] Ioan Stan and Răzvan Andonie. Cybernetical model of the artist-consumer relationship (in Romanian). *Studia Universitatis Babes-Bolyai*, 2:9–15, 1977.

[101] Chengyue Sun, Liang Wang, Xiaodong Liu, and Jingping Shi. Patient knowledge distillation for bert model compression. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3873–3879, 2019.

[102] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction (2nd Edition)*. MIT Press, 2018.

[103] Vivienne Sze, Yu-Hsin Chen, Tien-Ju Yang, and Joel S. Emer. Efficient processing of deep neural networks: A tutorial and survey. *CoRR*, abs/1703.09039, 2017.

[104] C Szegedy, W Liu, Y Jia, P Sermanet, SE Reed, D Anguelov, D Erhan, V Vanhoucke, and A Rabinovich. Going deeper with convolutions. *CoRR*, abs/1409.4842, 2014.

[105] Christian Szegedy, Sergey Ioffe, and Vincent Vanhoucke. Inception-v4, inception-resnet and the impact of residual connections on learning. *CoRR*, abs/1602.07261, 2016.

[106] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1701–1708. IEEE, 2014.

[107] M Tan and QV Le. EfficientNet: Rethinking model scaling for convolutional neural networks. *CoRR*, abs/1905.11946, 2019.

[108] Kumiko Tanaka-Ishii. *Semiotics of Programming*. Cambridge University Press, USA, 1st edition, 2010.

[109] Nicolás I. Tapia and Pablo A. Estévez. On the information plane of autoencoders. *CoRR*, abs/2005.07783, 2020.

[110] Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method. In *Proc. of the 37-th Annual Allerton Conference on Communication, Control and Computing*, pages 368–377, 1999.

[111] Naftali Tishby and Noga Zaslavsky. Deep learning and the information bottleneck principle. In *2015 IEEE Information Theory Workshop (ITW)*, pages 1–5, 2015.

[112] Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, ..., and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

[113] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017.

[114] E. Volden, G. Giraudon, and M. Berthod. Modelling image redundancy. In *1995 International Geoscience and Remote Sensing Symposium, IGARSS '95. Quantitative Remote Sensing for Science and Applications*, volume 3, pages 2148–2150, 1995.

[115] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, and Eftychios Protopapadakis. Deep learning for computer vision: A brief review. *Computational Intelligence and Neuroscience*, 2018:1–13, 02 2018.

[116] Xiaojie Wang, Chaoqun Wang, Xilin Chen, and Liqing Zhang. Symbolic adversarial learning. *Pattern Recognition*, 103:107260, 2020.

[117] Zi Wang, Chengcheng Li, and Xiangyang Wang. Convolutional neural network pruning with structural redundancy reduction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14913–14922, June 2021.

[118] Wei Wen, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Learning structured sparsity in deep neural networks. In *Advances in Neural Information Processing Systems*, pages 2074–2082, 2016.

[119] Kristoffer Wickstrøm, Sigurd Løkse, Michael Kampffmeyer, Shujian Yu, Jose Principe, and Robert Jenssen. Information plane analysis of deep neural networks via matrix-based Renyi's entropy and tensor kernels. 2019.

[120] Judith Williamson. *Decoding Advertisements: Ideology and Meaning in Advertising*. Marion Boyars Publishers, 1978.

[121] Andrew KC Wong and Mark A Vogel. Resolution-dependent information measures for image analysis. *IEEE Transactions on Systems, Man, and Cybernetics*, 7(1):49–61, 1977.

[122] Xiaofei Wu, Rongrong He, Zhenan Sun, and Tieniu Tan. A survey of compressing deep neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(3):705–723, 2021.

[123] Yinchong Yang, Zhenqiang Li, Xiaomeng Song, Chenghao Liu, Junhao Hou, and Lianli Gao. Interpretable convolutional neural networks. *arXiv preprint arXiv:1911.02508*, 2019.

[124] Quanming Yao, Mengshuo Wang, Hugo Jair Escalante, Isabelle Guyon, Yi-Qi Hu, Yu-Feng Li, Wei-Wei Tu, Qiang Yang, and Yang Yu. Taking human out of learning applications: A survey on automated machine learning. *CoRR*, abs/1810.13306, 2018.

[125] Tom Young, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. Recent trends in deep learning based natural language processing. 2018.

[126] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *CoRR*, volume abs/1311.2901, 2013.

[127] Heinz Zemanek. Semiotics and programming languages. *Communications of the ACM*, 9(3):139–143, 1966.

[128] Qinglong Zhang, Yifan Zhu, and Lei Zhang. Visual interpretability for deep learning: a survey. *Frontiers of Information Technology & Electronic Engineering*, 19(1):27–39, 2018.

[129] Qinglong Zhang, Yifan Zhu, and Lei Zhang. Visualizing and understanding convolutional networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 818–833, 2018.

[130] Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. *CoRR*, abs/1512.04150, 2015.