Pázmány Péter Katolikus Egyetem
Roska Tamás Műszaki és Természettudományi Doktori Iskola

Fülöp András

Konvolúciós neurális hálózatok optimalizálása

PhD Disszertáció

Témavezető:
Horváth András, Ph.D.

Társ-témavezető:
Csaba György, Ph.D.

2024

Bevezetés

Manapság számos összetett problémára találunk elfogadható algoritmikus megoldásokat, beleértve az NP-teljes problémákat is. A mélytanulási módszerek segítségével elérhettük vagy meghaladtuk az emberi teljesítményt számos területen.

A mélytanulási megoldások általában nagy mennyiségű adatot és lenyűgöző számítási kapacitást igényelnek, amit CPU-n vagy gyakrabban GPU-kon hajtanak végre. Ezért a mélytanulási modellek tanítása sok energiát igényel, és karbonlábnyoma egyre jelentősebbé válik.

Vegyük például az OpenAI által készített GPT-3-at, egy 175 milliárd paraméterű, autoregresszív nyelvi modellt, amely kiváló teljesítményt ér el különböző NLP problémákban. Becsült energiafogyasztása a képzés során 1287 MWh, szén-dioxid-kibocsátása pedig 552 tCO_2 volt.

Megállapíthatjuk, hogy még a ma sikeresen használt mélytanulási megoldások esetében is az energiafogyasztás nagyon fontos szempont, és egyes esetekben (pl. beágyazott rendszerek vagy okostelefonok esetében) alapvetően döntő tényező lehet.

Disszertációm tézisei

Első tézispont

Frekvenciatartományban implementáltam egy konvolúciós neurális hálót, amely nem használ semmilyen inverz Fourier-transzformációt egyik rétegében sem, beleértve a klasszifikációs részt is.

Bemutattam egy alternatív megvalósítást az aktivációs függvények frekvenciatartományban való alkalmazására, és bemutattam egy lehetséges megoldást az inverz Fourier-transzformáció kiküszöbölésére a klasszifikációs réteg előtt.

Neurális hálózatom architektúráját egy- és kétdimenziós adathalmazokon teszteltem, és összehasonlítottam olyan hasonló hálózati megvalósításokkal, amelyek inverz Fourier-transzformációt tartalmaznak. A javasolt keretrendszer hasonló vagy jobb pontosságot ért el az inverz Fourier-transzformáció számítási költsége nélkül. Az MNIST adatkészlet esetében az inverz FFT-t tartalmazó architektúra maximális pontossága körülbelül 6%-kal csökkent a időtartománybeli referenciához képest

(ahol a maximum 98,75% volt), míg az inverz FFT-t nem tartalmazó megoldásom maximális pontossága csak körülbelül 4%-kal esett vissza.

Számítási hatékonyság szempontjából modellem jelentősen csökkentette a szorzások számát. Egy 28x28 méretű bemenet esetén a Fourier-tartományban a szorzások száma 3136 volt, szemben az időtartományban lévő 7056-tal. Ez a számítási költség csökkenés, és a modellünk által elért hasonló pontosság bizonyítja megközelítésünk hatékonyságát.

Módszerek a frekvenciatartományban

Módszerem a konvolúció tételén alapul, amely a következő:

$$\mathcal{F}\{f \star g\} = \mathcal{F}\{f\} \cdot \mathcal{F}\{g\}, \tag{1}$$

ahol $\mathcal{F}$ jelöli a Fourier transzformációját az f és g függvényeknek, a $\star$ pedig a konvolúció operátora, míg $\cdot$ az elemenkénti szorzást jelenti.

A konvolúciós neurális hálózat első és legfontosabb része maga a konvolúció, ahol elemenként szorozzuk meg a képeket (vagy idősorokat) a konvolúciós kernelek megfelelő értékeivel, amelyeket a szorzás előtt a frekvenciatartományba transzformálhatunk. Ha kisebb filtereket használunk, mint a képek mérete vagy az idősorok hossza, akkor a transzformáció előtt nullákkal kell kiegészíteni a kerneleket, hogy a bemenettel azonos méretű mátrixokat kapjunk és elvégezhető legyen a pontonkénti szorzás művelete. Ebből kifolyólag annál több számítást takaríthatunk meg, minél nagyobb a filter mérete. Mivel azonban a hálózatom egésze a Fourier-tartományban van, egy másik technikát alkalmaztam, és közvetlenül a frekvenciatartományban hoztam létre a kerneleket, ahelyett, hogy azokat Fourier-transzformációval időtartományból transzformáltam volna, ezzel ugyanis megtakaríthatjuk a filterek transzformációjának költségét a tanítás során. Ebben az esetben a kernel mérete megegyezik a bemeneti adat méretével. Ezt a megközelítést alkalmaztam tehát a kísérleteimben. (Ez a megközelítés nincs hatással az inferencia idejére, ami a neurális hálózatok használata során egy fontos tényező, viszont csökkentheti a tanítási időt.)

Spektrális reprezentációban is különböző aktivációs függvény implementációkkal találkozhatunk, melyek célja, hogy hasonlóan működjenek, mint az időtartomány nemlinearitásai, és hasonló eredményt érjenek el pontosság szempontjából.

A Fourier-transzformáció során az eredeti bemeneti jelet a valós számok halmazából a komplex számsíkra visszük át, így az aktivációs függvényünk értelmezési tartománya is a komplex számok halmaza lesz.

Az én nemlineáris függvényem (FReLU) $f : \mathbb{C} \to \mathbb{C}$ az alábbi módon írható fel:

$$f(z) = \begin{cases} z & if \ \ |z| > \alpha \\ \mathbf{0} & \text{egyébként} \end{cases} \tag{2}$$

ahol $z \in \mathbb{C} = a + ib$ komplex számmal, a $|z| = \sqrt{a^2 + b^2}$, $\mathbf{0}$ pedig a $(0, 0)$ pontot jelenti a komplex síkon és az α egy hangolható paraméter. Ez a módszer tekinthető egy felül-áteresztő szűrőnek α vágási ponttal, vagy a hagyományos ReLU függvény komplex megfelelőjének. A Fig. 1 ábra illusztrálja, hogy ez a függvény hogyan képezi le a komplex síkot $\alpha = 0.1$ esetén. A tanításaim alatt az α paraméter értéke 0.1 volt.

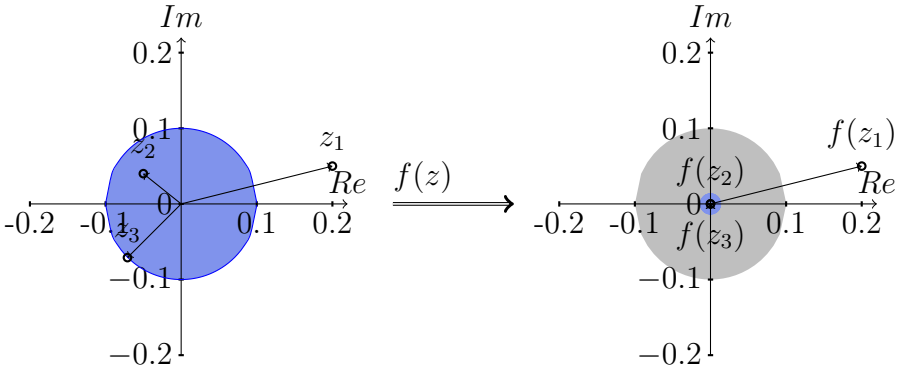

Figure 1: A nemlineáris aktivációs függvény f ($\alpha = 0.1$) leképzi z_1-et z_1-be és z_2-t, z_3-t pedig nullába, ahol $|z_1| > 0.1$, $|z_2| < 0.1$ és $|z_3| = 0.1$. A pontok pozíciója a kék körön kívül nem változik, míg az összes többi pont, ami a körben van nulla lesz.

A spektrális pooling módszert alkalmaztam mint alulmintavételezési eljárást. Ebben az esetben a dimenziócsökkentés a Fourier-tartományban történik, ahol az $N \times M$ mátrix bemenetet megvágjuk és csak a középső $H \times W$ méretű almátrixban lévő frekvenciák maradnak meg.

Az utolsó konvolúciós réteg után kiszámítjuk a komplex értékek nagyságát egy $f_{abs^2} : \mathbb{C} \to \mathbb{R}$ függvény alkalmazásával, amely a következőképpen írható le:

$$f_{abs^2}(a + ib) = a^2 + b^2 \tag{3}$$

Ez hasonló a korábban bemutatott aktivációs függvényhez, de a kimenet az abszolút érték négyzete, ami egy valós szám. Ennek a számításnak a számítási komplexitása $\mathcal{O}(n)$ az inverz FFT $\mathcal{O}(nlog(n))$ komplexitásával szemben. Ezután egy hagyományos teljesen összekapcsolt neurális hálózatot használtam egyetlen réteggel az osztályok meghatározásához.

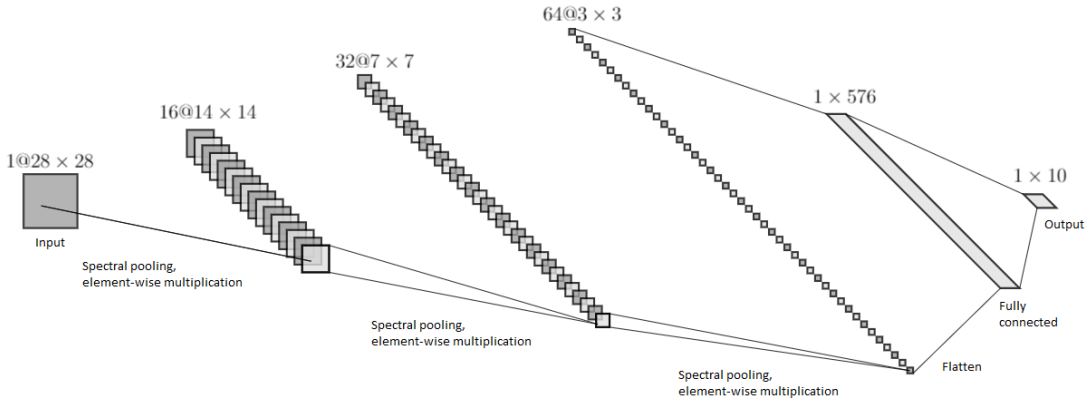

Figure 2: A javasolt CNN architektúrájának váza. A bemenet a frekvenciatartományban van, és a spektrális pooling elvégezhető az elemenkénti szorzás előtt, a nemlineáris aktivációs függvény pedig alkalmazható minden szorzás után.

Eredmények

Megvizsgáltam, hogy működik az architektúrám egy- és kétdimenziós adathalmazokon. Összehasonlítás céljából az időtartományban is megvalósítottam egy CNN-t, amelynek amennyire csak lehet ugyanaz a számítási komplexitása, mint a frekvenciatartományban működő neurális hálózatomnak (az időtartományban max poolingot és ReLU-t használtam). Ezeknek a CNN-eknek a különböző adathalmazokon elért pontossági eredményei a Table 1 táblázatban láthatóak.

Table 1: A táblázat tartalmazza az átlagos és maximális pontosságot (mindegyik esetben öt különböző tanítást végeztem és azokat átlagoltam illetve néztem meg azok maximumát), amit a vizsgált adathalmazok független tesztkészletein értem el három különböző hálózati architektúra esetében. Az egyik a referenciahálózat az időtartományban, a másik az inverz FFT-t tartalmazza, ahogy azt korábban is használták, valamint az én általam javasolt négyzetösszeg megoldással megvalósított neurális hálózat.

	inverse FFT		sum of squares		time domain	
Dataset	mean	max	mean	max	mean	max
MNIST	90.20%	92.39%	91.93%	94.99%	97.17%	**98.75%**
Fashion-MNIST	80.31%	81.95%	75.34%	82.83%	94.55%	**95.54%**
HADB	92.33%	94.08%	90.54%	93.95%	94.6%	**95.95%**
OZONE	90.26%	96.4%	96.07%	96.4%	94.31%	**97%**

Minden esetben öt különböző tanítást végeztem, és meghatároztam ezek maximumait, minimumait és átlagértékeit. Ezután összehasonlítottam az inverz FFT-t tartalmazó háló eredményeit a javasolt módszeremmel. Az MNIST, Fashion-MNIST és OZONE esetében azt találtam (lásd Table 1), hogy a maximum érték magasabb volt (vagy azonos az OZONE maximuma esetén) a javasolt módszeremmel, szemben az inverz FFT módszerével, és csak a HADB pontossága volt rosszabb. Azonban minden esetben csökkent a számítások száma, mivel a konvolúciós rétegek után nem $\mathcal{O}(nlog(n))$, hanem csak $\mathcal{O}(n)$ műveletet kellett végrehajtani (ahol n a minta mérete).

Bár az időtartománybeli neurális hálózat felülmúlta a két frekvenciaalapú megvalósítás pontosságát, ebben az esetben sokkal több szorzásra van szükség, mivel az időtartományban a konvolúció számítási komplexitása $\mathcal{O}(nm)$, ahol $m(= H \times W)$ a kernel mérete, de a frekvenciatartományban csak $\mathcal{O}(\frac{n}{4})$ komplexitással rendelkezünk, mivel a frekvenciatartományban a spektrális pooling elvégezhető az elemenkénti szorzás előtt. A frekvenciatartományban az FFT is számítást igényel ($\mathcal{O}(nlog(n))$), de ez elvégezhető (és tárolható) a tanítás előtt.

Második tézispont

Bemutattam egy új kernel konvolúciós neurális hálózatot, amely a felületi akusztikus hullámok elvei alapján, speciális konvolúciós eszközzel is megvalósítható.

Teszteltem neurális hálózatom architektúráját egy- és kétdimenziós adathalmazokon, és összehasonlítottam egy hasonló neurális hálózati megvalósítással, amely normál konvolúciót tartalmaz. A javasolt keretrendszerem hasonló vagy alig rosszabb pontosságot ért el, viszont potenciálisan sokkal gyorsabb és energiahatékonyabb eszközön implementálható.

Az MNIST adathalmazon a hálózatom átlagos pontossága 86,51%, maximális pontossága pedig 93,58% volt, szemben a referencia hálózat 92,61%-os átlagos és 96,52%-os maximális pontosságával. Hasonló tendenciákat figyeltem meg a Fashion-MNIST és HADB adathalmazokon is, ahol az átlagos teljesítmény körülbelül 6%-kal esett vissza.

Eredményeim feltárták a jövőbeli mágneses eszközök egy szükséges tulajdonságát is. Azt találtam, hogy a magas pontosság biztosítása érdekében az elnyelődési paraméter nem lehet rosszabb, mint $e^{\frac{-i}{999}}$

Módszerek

Javasoltam egy olyan speciális konvolúciós neurális hálózat architektúrát, amely nem tartalmaz klasszikus nemlineáris aktivációs függvényeket (mint például a ReLU), helyette a rendszer a szimulált eszköz fizikai tulajdonságain (elnyelődés és szaturáció) keresztül tartalmaz nemlinearitást a konvolúciós/kervolúciós rétegben.

A neurális hálózatunk megvalósítása során az elsődleges szempont az volt, hogy megvizsgáljam azokat a fizikai hatásokat, amelyeket egy olyan eszköz - amelyet kifejezetten a konvolúció műveletének végrehajtására fejlesztettek ki - gyakorolhat egy ideális mesterséges neurális hálózatra.

A valós idejű SAW konvolver, amely a neurális hálózati architektúrám megvalósításának kiindulópontja volt, csak egydimenziós bemeneteken képes konvolúciót végrehajtani.

Ezért a hardveres megközelítés miatt egydimenziós konvolúciós neurális hálózatot készítettem. Szimulációim során egy- és kétdimenziós adathalmazokat is

vizsgáltunk, így át kellett alakítanunk a 2D-s bemeneti adatokat és a konvolúciós kerneleket egydimenziós vektorokká.

Egy CNN egyik fő része a konvolúciós réteg. Az f és g függvények egydimenziós konvolúciója a következőképpen írható le:

$$f \star g = \int_{-\infty}^{+\infty} f(\tau)g(t-\tau)d\tau \tag{4}$$

Mivel a bemeneti jelünk véges, az f függvény értéke nulla egy bizonyos intervallumon kívül (például $[0, t]$). Így a konvolúciós integrál értéke is nulla ebben az intervallumban, így a képlet a következőképpen átírható:

$$[f * g](t) = \int_{0}^{t} f(\tau)g(t-\tau)d\tau \tag{5}$$

Ez a művelet megvalósítható valós idejű SAW konvolverekkel, mint például a háromportos rugalmas SAW konvolver (3) nemlineáris működés mellett.

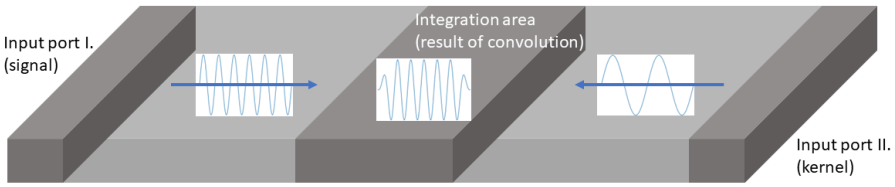

Figure 3: Ez az ábra egy primitív háromportos rugalmas SAW konvolvert ábrázol. Két jel ellentétes irányba utazik a készülék két bemeneti portjából (a készülék bal és jobb szélétől) és a két jel konvolvált változatát ki lehet nyerni az integrációs területen (a készülék közepén). Ez egy példa arra, hogy egy fizikai rendszert hogyan lehet felhasználni egy komplex művelet energiahatékony megvalósítására.

Ennek az eszköznek az első portján megy be a bemeneti jel, a túloldali portján a kernel, és ezek között van a harmadik port, a kimeneti vagy eredmény jel portja.

A bemeneti és kernel jeleket külső gerjesztéssel lehet bevezetni a készülék szélein, és a mágneses vagy elektromos változásokat ki lehet olvasni az eredmény portról.

Az Euler-képlet segítségével az I. port kifejezhető a t időpontban a z referencia tengely mentén a következőképpen:

$$s(t,z) = S(t - z/\nu)e^{j(\omega_0 t - \beta z)} \tag{6}$$

ahol $S(t - z/\nu)$ a jelmodulációs burkoló egy függvénye a SAW sebességnek, ahol $\nu = f\lambda$ és $\beta = 2\pi/\lambda$.

A 2. portot hasonlóképpen lehet felírni:

$$r(t,z) = R(t + z/\nu)e^{j(\omega_0 t - \beta z)} \tag{7}$$

ahol a z előjele negatív, mivel a jel az ellentétes irányba terjed.

Ebben az esetben a következő hullámformát lehet kiolvasni a kimeneti portról a vékony fémlemez L hosszán keresztül:

$$C(t) = P \int_{-\frac{L}{2}}^{+\frac{L}{2}} S(t - z/\nu)R(t + z/\nu)dze^{j2\omega_0 t} \tag{8}$$

ahol P egy konstans, amely a nemlineáris kölcsönhatás erősségétől függ. Használhatunk egy változócserét $\tau = (t - z/\nu)$ és átalakíthatjuk ezt az egyenletet a következőképpen:

$$C(t) = Mve^{j2\omega_0 t} \int_{-\infty}^{+\infty} S(\tau)R(2t - \tau)d\tau \tag{9}$$

ahol S a bemeneti jel, R a kernel jel, M egy a nemlineáris kölcsönhatás erejétől függő konstans, v a hullámok (jelek) sebessége, j a komplex egység és ω_0 a jel szögfrekvenciája.

A (4) és (9) egyenletek csak két tényezőben különböznek: a nemlineáris csillapításban ($Mve^{j2\omega_0 t}$) az egyenlet elején, és abban, hogy a kernel (R) argumentuma t helyett $2t$. Ennek az eltérésnek (időkompresszionak) az az oka, hogy a jelek egymás felé haladnak (relatív sebességük $2v$), így a kölcsönhatás feleannyi idő alatt zajlik le.

Számításaimban egy olyan eszközt vizsgáltam, amely hasonlóan működik, mint a valós idejű SAW konvolver, de a hullám erős csillapítást mutat - ezért a modell jól alkalmazható spin-hullám-szerű konvolverekre, ahol a csillapítás jelentősebb.

Az alap szimulációmban, egy négyzet jel ($s(t) = A_1 cos(\omega t)$) és egy háromszög jel ($r(t) = \frac{1}{t} A_2 cos(\omega t)$) haladnak egymással szemben, a hullámok nemlineáris módon terjednek. A négyzet és háromszög jeleket esettanulmányként választottam, mivel ezeket könnyű matematikailag leírni, és jól szemléltetik a konvolúció hatását.

A hullámok metszéspontjában kiolvasott jel a két bemeneti jel konvolúciója. (Valójában az egyik bemeneti jelet időben meg kell fordítani a konvolúció számításához, különben a jelek keresztkorrelációját kapjuk.) A szimuláció a 4. ábrán látható. A jel oszcilláló, de ha kihasználjuk azt a tényt, hogy a kimeneti jel frekvenciája kétszerese lesz az eredeti jelek frekvenciájának, akkor szűrhetjük a kimeneti jelet, és megkapjuk a konvolúció eredményét.

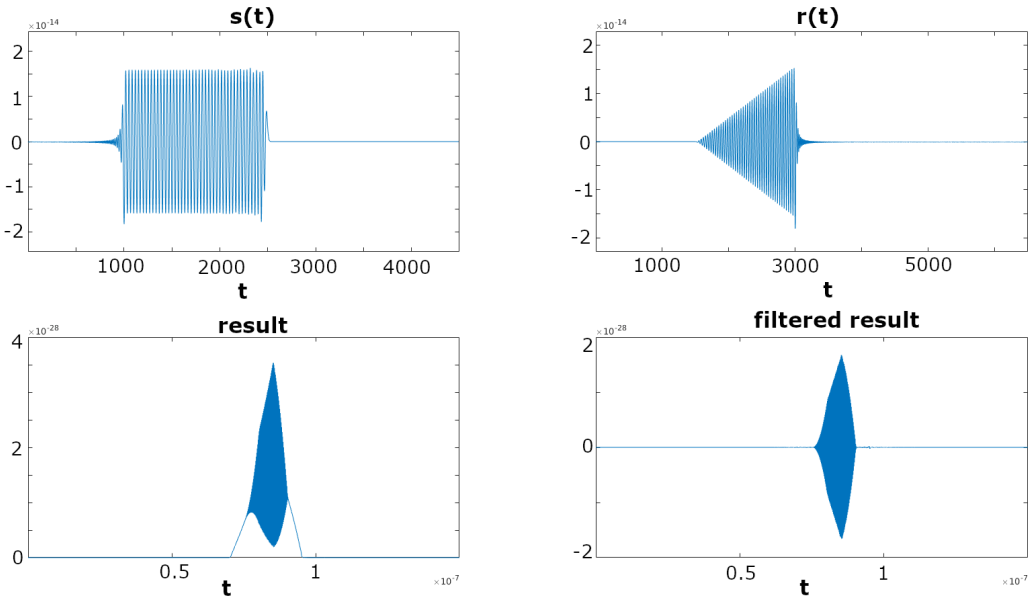

Figure 4: Az első sorban, $s(t)$ a négyzet jel, és mellette van a háromszög jel $r(t)$ (időben megfordítva), ezek egymásnak ellentétesen haladnak. A második sor első ábrája a nyers eredmény, amelyet a fenti jelek találkozásánál olvashatunk ki. Az utolsó ábra a frekvenciaszűrt eredmény, amelynek a frekvenciája kétszerese az eredeti jelek frekvenciájának.

A fizikai rendszerben a bemeneti jelek idővel csillapodnak, ahogy egyre távolabb haladnak a térben. Ezt a jelenséget figyelembe véve exponenciális csillapítást alkalmaztam mind a bemeneti jelre, mind a kernelre.

Fizikai rendszerünk tulajdonságainak megfelelően szaturációt alkalmaztam az elemenkénti szorzás után. Valójában a következő kernel konvolúciót használtam (a konvolúció i-edik eleme) CNN architektúrámban:

$$g_i(x) =< \phi_i(x_i), \phi_i(w) > \tag{10}$$

ahol $< \cdot, \cdot >$ két vektor skaláris szorzata hiperbolikus tangenssel (ami azt jelenti, hogy $< a, b >= \sum_{k=1}^{n} tanh(a_k b_k)$ és $tanh$ a rendszer telítettségét jelenti), és $\phi : \mathbb{R}^n \mapsto \mathbb{R}^n$ a következő nemlineáris függvény:

$$\phi_i(x) = e^{\frac{-i}{a}} x_i \tag{11}$$

ahol i a diszkrét idő, a a csillapítási paraméter. Az 5. ábra az $e^{\frac{-i}{a}}$ függvényt ábrázolja különböző a paraméterekkel. (Ezt a csillapítási képletet a következő módon is meg lehet fogalmazni: $0.999^i x_i$, ami $\phi_i(x)$, $a = 999$ paraméterrel.)

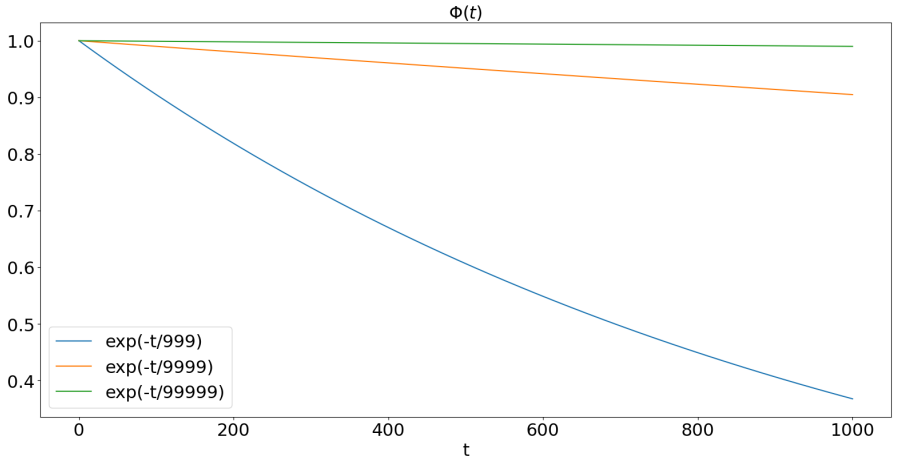

Figure 5: Szimulációm egyik paramétere a csillapítás. A terjedő hullámok exponenciálisan csillapodtak az $e^{\frac{-i}{a}}$ függvény szerint, ahol i az idő, és a a csillapítási paraméter. Ezen az ábrán, a függvényt 999, 9999 és 99999 csillapítási paraméterekkel ábrázoltam. A csillapítás mértéke jelentősen befolyásolhatja egy neurális hálózat pontosságát.

Eredmények

Egy egyszerű konvolúciós neurális hálózatból indultam ki, mellyel a célom az volt, hogy egy konvolúciós műveletet tartalmazó speciális fizikai architektúrát valósítsak meg, ami hatékonyan tudja végrehajtani a konvolúciót. Bevezettem fizikai jellemzőket a rendszerbe, hogy bemutassam ezeknek a jellemzőknek a hatásait. Ezután megvizsgáltam, hogyan működik az architektúrám (amelyet a 7. ábra mutat be) több egy- és kétdimenziós adatkészleten. Az összehasonlíthatóság érdekében implementáltam egy olyan CNN-t is, amely hasonló a neurális hálózatomhoz, de egydimenziós hagyományos konvolúciót használ. 1×9 kernellel és 3 réteggel használtam (két réteg 8 kernellel és egy réteg 16 kernellel), és minden konvolúciós réteg után ReLU-t alkalmaztam nemlineáris aktivációs függvényként a referencia CNN modellben, ahogy azt a 6. ábra mutatja. A hálózati architektúrám és a tanulási algoritmusok részletes paraméterbeállításai megtekinthetők a neurális hálózatom forráskódjában, amely a következő GitHub tárhelyen található: `https://github.com/andfulop/SpinWaveConvolver`. Ezen CNN-ek különböző adatkészleteken elért osztályozási pontosságai megtalálhatóak a 2. táblázatban.

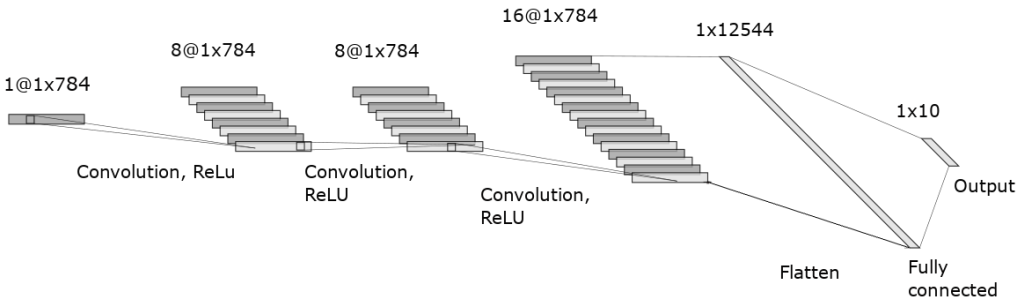

Figure 6: A referencia neurális hálózatom architektúrája. Hálózatom tartalmaz három konvolúciós réteget ReLU-kal, amelyeket egy teljesen összekapcsolt réteg követ. Ez az egyszerű négyrétegű architektúra képes egyszerű osztályozási feladatok megoldására.

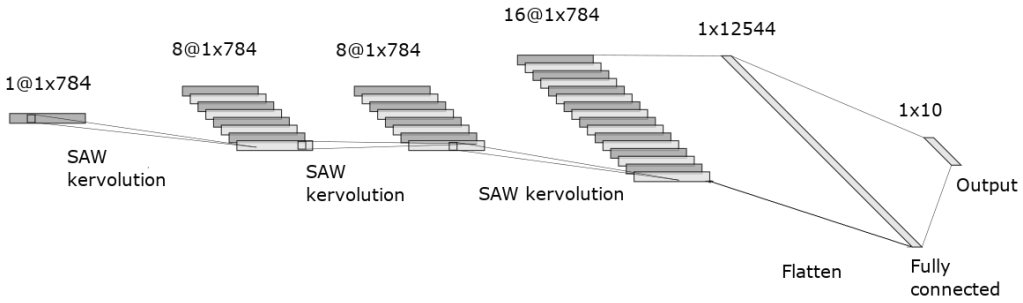

Figure 7: A neurális hálózatom architektúrája SAW kervolúcióval. Ebben a változatban a konvolúciókat és ReLU-kat kervolúciókkal helyettesítettem. A rétegek, csatornák és paraméterek száma mindkét hálózatban ugyanaz.

	Reference network		metwork with SAW kervolution	
Dataset	mean	max	mean	max
MNIST	92.61%	96.52%	86.51%	93.58%
Fashion-MNIST	77.84%	83.01%	72.87%	79.32%
HADB	88.43%	91.71%	82.11%	88.89%
OZONE	99.15%	99.2%	99.07%	99.4%

Table 2: Ez a táblázat a hagyományos konvolúciós hálózat (mint referencia) és az én módszerem, amely SAW konvolvert használ, pontosságait mutatja a különböző oszlopokban. A sorok négy különböző adatkészleten elért pontosságokat tartalmazzák. Ahogy az eredményekből látható, ugyanaz a hálózat különböző átlagos pontosságokat ért el különböző problémákon, 77-től 92%-ig terjedően a konkrét feladat bonyolultságától függően. Megfigyelhető egy körülbelül 6%-os teljesítménybeli csökkenés szinte minden esetben (kivéve az OZONE adatkészletet), és ez a csökkenés független a referencia hálózat eredeti pontosságától.

A eredmények azt mutatják, hogy a konvolúciót kervolúcióra lehet cserélni egy kb. 6%-os pontosság csökkenés mellett, ami lehetővé teheti az egyszerű neurális hálózatok energiatakarékos megvalósítását SAW konvolverrel. Sajnos egy ideális neurális hálózatban a jelek végtelen sebességgel terjednek, csillapítás és zaj nélkül. Annak bemutatására, hogy egy SAW elveit alapul vevő eszköz gyakorlati használhatósága milyen lehet, megvizsgáltam, hogy a különböző csillapítási

paraméterekkel rendelkező eszközök, hogyan teljesítenek az MNIST és HADB adatkészleteken.

Azt tapasztaltam, hogy ha a csillapítás mértéke túl nagy, akkor a hálózat pontossága jelentősen csökken, így a SAW alapú konvolverek fizikai tervezése során erre fokozottan ügyelni kell, olyan anyagokat és frekvenciákat kell választani, ahol biztosított a kismértékű csillapítás.

Összegzés

A disszertáció két új megközelítést mutat be a konvolúciós neurális hálózatok (CNN-ek) optimalizálására.

Az első megközelítés az egész tanítási folyamat frekvenciatartományban történő megvalósítására összpontosít, úgy, hogy nem tartalmaz inverz Fourier-transzformációt. A hagyományos aktivációs függvények frekvenciatartományban történő megvalósításának bevezetésével hasonló pontosságot értem el az inverz Fourier-transzformáció számítási költsége nélkül. A javasolt keretrendszert egy- és kétdimenziós adatkészleteken teszteltem, és az eredmények alátámasztják a hatékonyságot.

A második megközelítésben egy speciális CNN architektúrát vezettem be egy hullámalapú konvolverrel, amely fizikai megfontolásokon alapul. A klasszikus nemlineáris aktivációs függvények, mint például a ReLU vagy a szigmoid helyett, a rendszer a szimulált eszköz fizikai tulajdonságain keresztül építi be a nemlineáris jellemzőket, továbbá a fizikai jelenség természetében számolja ki a konvolúció műveletét, így rendkívül energiahatékony és gyors megvalósításokat eredményezhet. Ennek a megközelítésnek a pontossága valamivel alacsonyabb a hagyományos CNN-ekhez képest, azonban lehetőséget nyit alacsony energiafelhasználású megvalósításokra, például beágyazott rendszerek esetében.

Publikációk

Folyóiratcikkek

1. Fülöp, A., Csaba, G., and Horváth, A., "A Convolutional Neural Network with a Wave-Based Convolver." Electronics, 12(5), 1126, 2023.
2. Fülöp, A., Horváth, A., "End-to-end Training of Deep Neural Networks in the Fourier Domain.", MDPI Mathematics, 2022.

Konferenciacikkek

1. Fülöp, A., Horváth, A., "Application of Cellular Neural Networks in Semantic Segmentation.", IEEE International Symposium on Circuits and Systems, 2021.
2. Fülöp, A., Horváth, A., "Template Optimization in Cellular Neural Networks Using Gradient Based Approaches." 2020 European Conference on Circuit Theory and Design (ECCTD). IEEE, 2020.