

TDK-DOLGOZAT

SZERZŐ: ÁCS HUNOR

XXVII. REÁL- ÉS HUMÁNTUDOMÁNYI ERDÉLYI TUDOMÁ-
NYOS DIÁKKÖRI KONFERENCIA (ETDK)

INFORMATIKA I: ELMÉLETI MÓDSZEREK, KÍSÉRLETEK,
SZIMULÁCIÓK, MATEMATIKA ÉS MÓDSZERTAN

A tudásdesztilláció hatékonysága különböző méretű konvolúciós neurális hálózatok esetében

SZERZŐ: ÁCS HUNOR

BABEŞ-BOLYAI TUDOMÁNYEGYETEM, MATEMATIKA ÉS INFOR-
MATIKA KAR, INFORMATIKA SZAK, III. ÉV

TÉMAVEZETŐ: DRD. PÁVEL SZABOLCS, TANÁRSEGÉD

BABEŞ-BOLYAI TUDOMÁNYEGYETEM, MATEMATIKA ÉS INFOR-
MATIKA KAR

KOLOZSVÁR, 2024

Tartalomjegyzék

1. Bevezető	1
1.1. Motiváció	1
1.2. Kutatási terület	2
1.2.1. Alkalmazási területek	2
1.2.2. Kihívások és korlátok	3
2. Elméleti alapok	4
2.1. Gépi tanulás	4
2.1.1. Történelmi háttér	4
2.1.2. Mélytanulás	5
2.1.3. Neurális hálózatok fejlődése	5
2.1.4. Konvolúciós neurális hálózatok	7
2.1.5. Optimalizálási algoritmusok	8
2.2. Arhitektúrák	9
2.2.1. ResNet	9
2.2.2. MobileNet	11
2.2.3. Big Transfer (BiT)	12
2.3. Tudás desztilláció	13
2.3.1. A jó tanár türelmes és következetes	14
2.3.2. További tudás desztilláció megközelítések	14
3. Saját kísérletek	16
3.1. Használt arhitektúrák	16
3.1.1. Tanár	16
3.1.2. Diák	17
3.2. Adathalmazok	18
3.2.1. ImageNet1K	18
3.2.2. OxfordFlowers102	18
3.3. Tanítási módszerek	19
3.3.1. Tanulási stratégiák	19
3.3.2. Optimalizációs Algoritmusok	21
3.3.3. Értékelési metrikák és validálás	21
3.3.4. Kísérleti környezet	22
3.4. Eredmények	22

TARTALOMJEGYZÉK

3.4.1.	Konfigurációk és kvantitatív eredmények	23
3.4.2.	Eredmények elemzése	24
4.	Konklúzió	27
4.1.	További kutatási lehetőségek	27
5.	Köszönetnyilvánítás	29
A.	Felhasznált eszközök	33

1. fejezet

Bevezető

Összefoglaló: A tudás desztilláció egy olyan technika, amelynek célja a nagy, bonyolult neurális hálózatok által megszerzett tudás átadása egy kisebb, egyszerűbb modellnek, hogy javuljon annak teljesítménye, miközben csökkentjük a szükséges számítási erőforrásokat. A jelen kutatás a tanulási eredmények és a modell méretének kapcsolatában vizsgálja e technika hatékonyságát, különös tekintettel a különböző méretű, szokásosnál kisebb konvolúciós hálókra. Célunk a desztillációs folyamat eredményességének befolyásolásának feltárása a hálók mérete szerint, hogy jobban megértsük a tudás desztillációjának méretfüggő hatékonyságát. Kutatásunk jelentős hozzájárulás lehet az effektív modellkompressziós stratégiák fejlesztéséhez, különösen számítási vagy tárhely-kapacitás szempontjából korlátozott környezetek számára.

1.1. Motiváció

Az elmúlt évtizedben a mélytanulási modellek és algoritmusok exponenciális fejlődésen mentek keresztül, jelentős mértékben növelve azok hatékonyságát számos feladattípuson, mint például képfelismerés, természetes nyelvfeldolgozás és játékontelligencia. Ezek a modellek azonban jelentős számítási erőforrásokat igényelnek mind a tanítás, mind az alkalmazás során, ami korlátozza azok használhatóságát számítási kapacitással vagy energiaellátással korlátozott környezetekben, mint például mobileszközökön, beágyazott rendszereken, vagy távoli érzékelési alkalmazásokban.

A tudás desztilláció (knowledge distillation) azon technikák csoportjába tartozik, amelyek lehetővé teszik a nagy, összetett modellek által tanult információk átadását kisebb, kevésbé erőforrás-igényes modellekbe. Ez a technika javíthatja a kisebb modellek teljesítményét, lehetővé téve számukra, hogy meglepően jól teljesítsenek szűkös számítási körülmények között.

E kutatás alapvető motivációja, hogy feltárja a tudás desztilláció hatékonyságát különböző méretű és architektúrájú modellek esetén. Különös figyelmet fordítunk a konvolúciós neurális hálózatok (CNN) alkalmazására, amelyek széles körben használatosak a képfeldolgozás terüle-

1. FEJEZET: BEVEZETŐ

tén. Az ezen technikával kapcsolatos korábbi kutatások többsége a nagy modellek teljesítményének reprodukálására koncentrált kisebb modellekben, viszont kevésbé tárgyalta a különböző architektúrák és méretek közötti teljesítménybeli különbségeket és azok okait.

A kutatás célja, hogy mélyebb betekintést nyújtson a desztillációs folyamat méretfüggő hatékonyságába és annak optimalizálásába. A cél az, hogy fejlesztési javaslatokat tegyünk az effektív modellezési és tanítási stratégiák tervezéséhez, különösen azokban az esetekben, ahol a számítási erőforrások vagy a tárolási kapacitás szűkössége jelent korlátozó tényezőt.

Ezen felül a kutatásunk lehetőséget kínál arra, hogy összehasonlítsuk a különböző méretű modellek közötti tudásátadás hatékonyságát, és megállapítsuk, hogy milyen mértékben befolyásolja a modell komplexitása és a tanítási módszerek a desztillációs folyamat sikerét. Az eredmények alapján javaslatokat tehetünk arra vonatkozóan, hogy milyen architektúrák és paraméterezési stratégiák preferáltak különböző felhasználási esetekben, amelyek széles körű alkalmazást találhatnak az iparban és az akadémiai kutatásokban egyaránt.

1.2. Kutatási terület

A gépi tanulás és mélytanulás területén az egyik legfontosabb cél a nagy teljesítményű modellek komplexitásának és erőforrás-igényének csökkentése. A tudás desztilláció (knowledge distillation) technika éppen erre a problémára kínál megoldást: egy nagy, bonyolult "tanár" modell tudását átviszi egy kisebb, egyszerűbb "diák" modellbe.

A tudás desztilláció működése során a diák modell igyekszik utánozni a tanár modell kimeneti valószínűségi eloszlását, ami általában a logitok (a modell kimeneti rétege előtti aktivációk) "lágyításával" történik, ahol a lágyítást egy hőmérsékleti paraméter szabályozza. A desztilláció során a diák modell nem csak az egyes adatpontokra adott konkrét válaszokat tanulja meg, hanem a tanár modell általánosítási képességét is igyekszik átvenni.

1.2.1. Alkalmazási területek

A tudás desztilláció alkalmazható különböző iparágakban és szektorokban, ahol a modell méretének vagy a számítási igényeknek korlátozottak:

- **mobil eszközök és beágyazott rendszerek:** itt a hely és az energiaforrás korlátozott, így a kisebb, hatékonyabb modellek előnyösek;

1. FEJEZET: BEVEZETŐ

- **felhő alapú szolgáltatások:** kisebb modellekkel csökkenthető a szerver oldali terhelés, így a szolgáltatások gyorsabban és költséghatékonyabban működhetnek;
- **kiterjesztett valóság (AR) és virtuális valóság (VR):** itt a valós idejű adatfeldolgozás kritikus, így a gyors és hatékony modellek kulcsfontosságúak.

1.2.2. Kihívások és korlátok

Bár a tudás desztilláció ígéretes megoldást kínál a modell kompresszióra, több kihívással is szembesül:

- **tudásátadás hatékonysága:** nem minden esetben nyilvánvaló, hogy melyik tanár-diák modell párosítás lesz a legjobb. Néha a diák modell nem képes teljesen megtanulni a tanár modell által nyújtott tudást;
- **optimális hőmérsékleti paraméter meghatározása:** a desztillációs folyamat során a hőmérsékleti paraméter kulcsfontosságú, de annak optimális értékének megtalálása nem triviális.
- **különböző adathalmazok és feladatok:** az adathalmazok és a tanulási feladatok változatossága tovább bonyolítja a megfelelő modellek kiválasztását és a desztillációs folyamatot.

2. fejezet

Elméleti alapok

2.1. Gépi tanulás

A gépi tanulás a mesterséges intelligencia azon alterülete, amely olyan számítógépes rendszerek létrehozására összpontosít, amelyek képesek adatokból tanulni, és javítani önmagukon.

A gépi tanuláson alapuló algoritmusok manapság rendkívül elterjedtek, számos iparágat átfognak: a pénzügyi szektortól kezdve az egészségügyön át a kiskereskedelmen, gyártáson, autóiparon, média és szórakoztatáson, oktatáson, energetikán, mezőgazdaságon, egészen a közlekedés és logisztikáig széles körben alkalmazzák. Ezek az algoritmusok lehetővé teszik a csalásfelismerést, betegségek korai felismerését, személyre szabott kezeléseket, árazási stratégiák optimalizálását, termelési folyamatok fejlesztését, önvezető járművek fejlesztését, tartalomajánló rendszerek kifejlesztését, tanulási folyamatok személyre szabását, energiafogyasztás előrejelzését, terméshozam előrejelzését, és sok más területen javítják az hatéktivitást, pontosságot, nyitva új lehetőségeket az innovációra és fejlődésre.

2.1.1. Történelmi háttér

A gépi tanulás tudományának története 1943-ban vette kezdetét, amikor Walter Pitts és Warren McCulloch "A logical calculus of the ideas immanent in nervous activity" című munkájukkal új alapokra helyezték az idegrendszer működésének matematikai modellezését [Pitts és McCulloch, 1943]. Nem sokkal később, 1949-ben Donald Hebb "The Organization of Behavior" című könyvével tovább bővítette a tudományág alapjait, amelyben bemutatta az idegsejtek közötti kölcsönhatások alapján kialakuló struktúrákat, ismertebb nevén a Hebbi tanulást [Hebb, 1949]. Ez a mű alapvető fontosságú volt az agyi tanulási folyamatok megértésében, és közvetett módon hozzájárult a gépi tanulási algoritmusok fejlődéséhez.

2.1.2. Mélytanulás

A mélytanulás, a gépi tanulás egyik specializált ága, többrétegű neurális hálózatokat, azaz mély neurális hálózatokat alkalmaz az emberi agy döntéshozatali képességének modellezésére. Ez a technika arra tanítja a számítógépeket, hogy az adatokat az emberi agyhoz hasonlóan dolgozzák fel, lehetővé téve összetett mintázatok felismerését képekben, szövegekben, hangokban és egyéb adatformátumokban, pontos betekintések és előrejelzések nyújtásával. Továbbá ez a napjaink szinte minden mesterséges intelligenciáját meghajtó technológiája, lehetővé téve az eszközök számára, hogy olyan feladatokat hajtsanak végre, amelyek korábban kizárólag az emberi intelligencia privilégiumai voltak.

1958-ban került bemutatásra a perceptron modell Frank Rosenblatt által, egy mesterséges neuronokon alapuló algoritmus [Rosenblatt, 1958]. A perceptron képes volt tapasztalatokból tanulni és ezek alapján módosítani a súlyait a pontosabb előrejelzések érdekében. Azonban a perceptronok csak lineárisan szeparálható problémák megoldására voltak képesek, korlátjaik pedig csökkentették az érdeklődést a neurális hálózatok iránt. Ezután következett be az első "tél" a mesterséges intelligencia területén, amely olyan időszakot jelölt, amikor kevés vagy semmilyen előrelépés nem történt a területen, a kutatási beruházások hiánya miatt. A mélytanulás teljesen kihalt az 1960-as évek végén, mivel a kutatók nem voltak meggyőződve a bináris neuronokról.

A kudarc után számos kutatás történt, és végül 1986-ban Geoffrey Hinton, David Rumelhart és Ronald Williams bemutatták a visszaterjesztési(backpropagation) algoritmust [Rumelhart et al., 1986]. A visszaterjesztés lehetővé tette a többrétegű neurális hálózatok képzését a hálózat súlyainak gradiensének hatékony kiszámításával. Ez újra felkeltette az érdeklődést a neurális hálózatok iránt és megnyitotta az utat a mélytanulás fejlesztése felé. Hinton személyét gyakran nevezik a mélytanulás atyjának.

2.1.3. Neurális hálózatok fejlődése

Az 1990-es évek végén, elsősorban Yann LeCunnak köszönhetően létrejött a Konvolúciós neurális háló (CNN) [LeCun et al., 1998]. LeCun kifejlesztette a LeNet-5 architektúrát, amely konvolúciós rétegeket, pooling rétegeket és teljesen összekapcsolt rétegeket alkalmazott a kézzel írt számjegyek felismerésében. A CNN-ek forradalmasították a számítógépes látást azáltal, hogy lehetővé tették a képek jellemzőinek kinyerését, ami áttörést eredményezett a képkosztá-

2. FEJEZET: ELMÉLETI ALAPOK

lyozásban, objektumfelismerésben és egyebekben.

A 2000-es évektől kezdve a mélytanulás területe robbanásszerű fejlődésen ment keresztül, amelynek köszönhetően jelentősen megnőtt a mesterséges intelligencián alapuló alkalmazások száma és sokfélesége. Ebben az időszakban számos technológiai és elméleti áttörés történt, amelyek lehetővé tették a mélytanulási modellek hatékonyságának és alkalmazhatóságának jelentős növekedését.

A 2000-es évek elejét egyfajta adatforradalom jellemezte, amikor is az adatok mennyisége exponenciálisan növekedni kezdett. Ez összefüggésbe hozható az internet és a közösségi média terjedésével, amely hatalmas adathalmazokat generált. Emellett a hardverek, különösen a grafikus feldolgozó egységek (GPU-k) fejlődése lehetővé tette a nagy mennyiségű adatok gyors feldolgozását, így jelentősen csökkentve a mélytanulási modellek tanításához szükséges időt.

A 2010-es évek jelentős áttöréseket hoztak a mélytanulás terén. A konvolúciós neurális hálózatok (CNN-ek) és a rekurrens neurális hálózatok (RNN-ek) terén elért fejlesztések lehetővé tették a számítógépes látás és a természetes nyelvfeldolgozás (NLP) előrelépéseit. Az ImageNet versenyen elért sikerek, különösen a 2012-es AlexNet általi győzelem [Krizhevsky et al., 2012], világszerte felhívták a figyelmet a mélytanulásra mint egy rendkívül ígéretes MI technológiára.

A mélytanulási modellek alkalmazása exponenciálisan növekedett az évtized során, kezdve az arcfelismeréstől, beszédfelismerésen át, a fordítóprogramokig, és az autonóm járművekig. Ezek a modellek képesek voltak már felülmúlni az embert bizonyos feladatokban, mint például a Go játékban ([Wang et al., 2016]) vagy a bőrdaganatok diagnosztizálásában ([Li et al., 2020]).

A 2020-as évekre a mélytanulás már szerves részévé vált mindennapi életünknek, beleértve az okoseszközöket, a személyi asszisztenseket és a személyre szabott ajánlórendszereket. A Generatív Adversarial Networks ([Goodfellow et al., 2020]) és a Transformer modellek, mint például a GPT-3 ([Floridi és Chiriatti, 2020]), új lehetőségeket nyitottak a tartalomgenerálás, a művészet, a tervezés és az interaktív szórakoztatás terén, ami főként Ashish Vaswani és társai "Attention Is All You Need" című munkájának köszönhető [Vaswani et al., 2017].

Bár a mélytanulás hatalmas előrelépéseket hozott, számos kihívással is szembesülünk, beleértve az adatok elfogultságát, az értelmezhetőséget, a magas energiafogyasztást és az MI etikai kérdéseit. A jövőbeli kutatások várhatóan ezekre a problémákra is megoldásokat keresnek, miközben folyamatosan bővíteni fogják a mélytanulás alkalmazási körét és javítják annak hatékonyságát.

2.1.4. Konvolúciós neurális hálózatok

A konvolúciós neurális hálózatok (CNN-ek) különleges típusú mély neurális hálózatok, amelyek kifejezetten képek feldolgozására lettek tervezve. Ezek a hálózatok az emberi látórendszer működését modellezzik, ezáltal felismerik a képekben található jellemzőket, mint például az éleket, a szögeket és a textúrákat. A konvolúciós architektúra kulcsfontosságú elemei a konvolúciós rétegek, az aktivációs függvények, a pooling rétegek és a teljesen összekapcsolt rétegek.

A konvolúciós rétegek alapvetően úgy működnek, mint egy speciális szűrőrendszer a képfeldolgozásban. Amikor egy képet beviszünk a hálózatba, ezek a rétegek segítenek kiszűrni és kiemelni a fontos jellemzőket, mint például az élek, színek vagy textúrák. Ez úgy történik, hogy a konvolúciós réteg több kis mátrixot, úgynevezett kernel-eket vagy szűrőket használ, amik áthaladnak a kép minden egyes részén. Ez a folyamat lényegében egy matematikai művelet, ami összeszorozza a kernel értékeit a kép adott részének pixel értékeivel, így létrehozva új képi jellemzőket, amiket a hálózat a tanulási folyamat során használ fel. Ez a módszer segít abban, hogy a modell jobban értelmettesse és felismerje a különböző formákat és objektumokat a képeken keresztül.

Az aktivációs funkciók, mint például a ReLU (Rectified Linear Unit), nemlineáris tulajdonságokat adnak a hálózathoz, lehetővé téve komplex jellemzők tanulását.

A pooling rétegek, köztük a leggyakrabban használt max pooling, csökkentik a térbeli méreteket azáltal, hogy összegzik a jellemzők részleteit, ezáltal csökkentve a szükséges számítások számát és javítva a modell általánosító képességét.

A teljesen összekapcsolt rétegek a hálózat végén helyezkednek el, és a jellemzőkből kinyert információk alapján hajtanak végre klasszifikációt vagy regressziót.

A CNN-eket széles körben alkalmazzák a számítógépes látás területén, beleértve az arcfelismerést, a járművek azonosítását a műholdképeken, a radiológiai képek elemzését és még sok mást.

A konvolúciós architektúrák fejlődése nem állt meg az AlexNetnél; számos továbbfejlesztett modell került bemutatásra, mint például a VGG [Simonyan és Zisserman, 2015], GoogLeNet [Szegedy et al., 2015], ResNet [He et al., 2016]. Ezek a modellek lehetővé tették a mélytanulási modellek egyre szélesebb körű alkalmazását és a pontosabb eredmények elérését a különböző feladatokban.

A konvolúciós neurális hálózatok sikere és fejlődése jól mutatja a mélytanulás területének

robusztus növekedését és adaptációját a változó igényekhez. Ahogy a technológia tovább fejlődik, várhatóan új architektúrák és módszerek jelennek meg, amelyek még jobban kiaknázzák a mélytanulás potenciálját a számítógépes látás és túlmutató területeken.

2.1.5. Optimalizálási algoritmusok

Az optimalizálási algoritmusok kulcsfontosságú szerepet játszanak a mélytanulásban, mivel ezek segítségével igyekszünk minimalizálni (vagy maximalizálni) a tanítási folyamat során a veszteségfüggvény (cost function) értékét, így javítva a neurális hálózat teljesítményét. Két népszerű és széles körben használt optimalizálási algoritmus a Stochastic Gradient Descent (SGD) és az Adam optimizer.

Stochastic Gradient Descent

Az SGD a Gradient Descent (GD) egy olyan változata, amely a tanító adatkészlet véletlenszerű mintáin, azaz batch-eken végzi el a gradiens lejtési lépéseket [Robbins és Monro, 1951]. Míg a hagyományos GD az összes tanító adaton kiszámítja a gradienst minden iterációban, ami nagyon időigényes lehet nagy adatkészletek esetén, az SGD csak egy véletlenszerűen kiválasztott részhalmazon végzi el ezt a számítást. Ez jelentősen felgyorsítja a konvergenciát és csökkenti a számítási költségeket, különösen nagy adatkészletek esetén. Azonban az SGD hajlamos lehet a zajos gradiens frissítésekre, ami változó konvergencia sebességet eredményezhet.

Adam

Az Adam (Adaptive Moment Estimation) egy viszonylag újabb optimalizálási algoritmus, amely adaptív tanulási ráta módszereket kombinál az első és második momentumok becslésével [Kingma és Ba, 2014]. Ezáltal az Adam képes alkalmazkodni a gradiensnek az egyes súlyokhoz való érzékenységéhez; nagyobb lépéseket tesz azoknál a súlyoknál, ahol a gradiens kicsi, és kisebb lépéseket azoknál, ahol a gradiens nagy. Ez a megközelítés javítja a konvergencia sebességét és stabilitását számos probléma esetén, különösen a mélytanulási alkalmazásokban, ahol a veszteségfüggvény komplex és a paraméterek száma hatalmas. Az Adam népszerűségét annak hatékonysága és rugalmassága magyarázza, mivel jól működik a legtöbb esetben anélkül, hogy a hiperparaméterek finomhangolására lenne szükség.

2. FEJEZET: ELMÉLETI ALAPOK

Mindkét algoritmusnak megvannak a maga előnyei és hátrányai. Az SGD egyszerűsége miatt továbbra is népszerű választás sok alkalmazásban, különösen mikor a zajos gradiens frissítések előnyös lehetnek a lokális minimumokból való kilépés szempontjából. Ezzel szemben az Adam adaptív tanulási rátái jobb konvergenciát és stabilabb tanulást biztosíthatnak, különösen mély neurális hálózatok esetében. A választás gyakran a konkrét alkalmazás sajátosságaitól és a kutató preferenciáitól függ.

A felsorolt optimalizációs algoritmusok, ötvözve a konvolúciós neurális hálózatokkal és más mélytanulási architektúrákkal, alapvetően változtatták meg azt, hogy hogyan dolgozunk fel és értelmezzük a nagy adathalmazokat, utat nyitva újítások és fejlesztések felé számos területen, a számítógépes látástól kezdve a természetes nyelvfeldolgozáson át, az egészségügyi diagnosztikáig és túl. Ahogy a kutatások tovább haladnak, és új technológiák fejlődnek, várhatóan további áttörések történnek majd, amelyek még több lehetőséget nyitnak meg a mesterséges intelligencia alkalmazásaiban. Az előttünk álló kihívások ellenére a gépi tanulás és a mélytanulás területén elért eddigi sikerek bizakodásra adnak okot a jövőbeli innovációk és áttörések tekintetében.

2.2. Arhitektúrák

2.2.1. ResNet

A 2015-ben publikált "Deep Residual Learning for Image Recognition" cikk bemutatta a ResNet architektúrát [He et al., 2016]. Ez az innováció az ImageNet Nagyméretű Vizuális Felismerési Kihíváson (ILVRSC) mutatkozott be először, ahol rendkívüli teljesítményével, mindössze 3,57%-os hibaarányt elérve, a top5 kategóriában győzedelmeskedett. A reziduális hálózatok előtt a konvolúciós hálók számos problémával küzdöttek.

- **Eltűnő/Robbanó gradiens:** ez a jelenség egy ismert probléma, amely során a gradiens értékei időnként túlzottan magasak vagy alacsonyak lehetnek. Ez azt eredményezte, hogy amikor a modell rétegeinek számát növelték, a hiba aránya is emelkedett;
- **Neurális hálózatok degradációja:** ennek problémájának két fő aspektusa van, amelyek közösen magyarázzák, miért nem mindig vezet jobb teljesítményhez a hálózat mélységének növelése. Elsőként, a teljesítmény plató jelenségével találkozhatunk, ami azt jelenti,

2. FEJEZET: ELMÉLETI ALAPOK

hogy ahogy a neurális hálózat rétegeinek száma növekszik, a tanulási hiba hajlamos a telítődésre és nem mutat további javulást. Ez azt sugallja, hogy a hozzáadott rétegek már nem képesek jelentős mértékben csökkenteni a tanulási hibát, tehát nem növelik a hálózat hatékonyságát. Második jelenséggént a pontosság romlása figyelhető meg, amikor további rétegek hozzáadása után váratlanul növekszik a validációs halmazon mért hiba, és a hálózat teljesítménye a látatlan adatokon gyengül. Ez arra utal, hogy a hálózat túlilleszkedik a tanulási adatokra, ami csökkenti az általánosító képességét és így rosszabb teljesítményt eredményez új, eddig nem látott adatokon.

A Microsoft kutatói 2015-ben válaszokat kutattak ezekre a kihívásokra, amikor is bemutatták a Reziduális Hálózatokat, amelyek számos jelentős újítást tartalmaztak:

1. **Reziduális blokkok:** a ResNet alapvető elemei, amelyek konvolúciós rétegekből, kötegelt normalizálásból és ReLU aktivációs függvényekből állnak;
2. **Áthidaló kapcsolatok (Skip Connections):** lehetővé teszik a bemenet közvetlen hozzáadását a blokk kimenetéhez, "átugorva" ezzel a rétegeket;
3. **Kötegelt normalizálás (Batch Normalization, [Ioffe és Szegedy, 2015]):** normalizálja a rétegek bemeneteit, stabilizálva a tanítási folyamatot;
4. **Identitás leképezések (Identity Mapping):** az áthidaló kapcsolatok alkalmazása változtatás nélkül, ha a dimenziók megegyeznek;
5. **Bottleneck architektúra:** három rétegből álló blokkok a számítási komplexitás csökkentésére;
6. **Global average pooling:** a jellemző térképek egyetlen számmá való redukálása az utolsó konvolúciós réteg után.

A Reziduális Hálózatok (ResNet) bevezetése jelentős mérföldkőnek számított a mélytanulás területén, különösen a képfelismerési feladatokban. Ezek az innovációk lehetővé tették a hálózatok mélységének növelését anélkül, hogy a tanulási folyamat során az eltűnő vagy robbanó gradiens problémájával, illetve a degradáció jelenségével kellene szembesülni. Az áthidaló kapcsolatok, kötegelt normalizálás és az identitás leképezések bevezetése révén a ResNet hálózatok képesek voltak mélyebb, mégis hatékonyan tanuló modelleket létrehozni. A jelentős javulás az

ImageNet ([Deng et al., 2009]) versenyen elért eredményekben is megmutatkozott, ahol a ResNet architektúra új standardokat állított fel. Ezek a fejlesztések nem csupán a képfelismerés, hanem a mélytanulás számos más területén is új lehetőségeket nyitottak meg, bizonyítva ezzel a mélyebb neurális hálózatok potenciálját és sokoldalúságát.

2.2.2. MobileNet

A MobileNet architektúrák célja, hogy hatékony architektúrát nyújtsanak olyan endszerek számára, ahol a számítási kapacitás és az energiafogyasztás korlátozott. A MobileNet modell különlegessége, hogy könnyűsúlyú konvolúciós hálózatokat használ, amelyek lehetővé teszik a gyors és hatékony előrejelzéseket még alacsony teljesítményű eszközökön is. [Howard et al., 2017] bemutatta az első MobileNet architektúrát, amely mélységi szeparálható konvolúciókra és a modell skálázhatóságának innovatív alkalmazására épült.

A MobileNet architektúrák alapvető jellemzői közé tartozik:

- **Mélységi azeparálható konvolúciók (Depthwise Separable Convolutions):** ez a technika két részre osztja a konvolúciós műveleteket: egy mélységi konvolúcióra, amely külön-külön alkalmazza a szűrőket minden bemeneti csatornára, és egy 1×1 -es konvolúcióra (pontoszerű konvolúció), amely összekapcsolja a csatornákat. Ez jelentősen csökkenti a számítási igényeket és a modell méretét anélkül, hogy jelentős teljesítménybeli kompromisszumot kellene vállalni;
- **Szélességi multiplikátor (Width Multiplier):** egy hiperparaméter, amely lehetővé teszi a modell méretének és számítási igényének dinamikus skálázását, csökkentve vagy növelve a csatornák számát a konvolúciós rétegekben;
- **Felbontás Multiplikátor:** lehetőséget biztosít a bemeneti képek felbontásának skálázására, lehetővé téve a számítási igények és a teljesítmény közötti finomhangolást.

A [Sandler et al., 2018] cikk a MobileNetV2-t bemutatva továbbfejlesztette az architektúrát az inverz reziduális szerkezetek és lineáris szűkületek bevezetésével. [Howard et al., 2019] a MobileNetV3 kutatásában pedig a legújabb technológiákat és technikákat integrálta a hatékonyság és teljesítmény további növelése érdekében. A MobileNet modell bevezetésével a Google újraértelmezte, hogy miként lehet magas szintű képfelismerési teljesítményt elérni korlátozott

erőforrásokkal rendelkező eszközökön. Ez a megközelítés új távlatokat nyitott meg a MI alkalmazásokban, lehetővé téve a fejlett képfeldolgozást és gépi látást még az olyan eszközökön is, ahol a számítási kapacitás és az energia rendkívül korlátozott. A MobileNet architektúrák által nyújtott rugalmasság és skálázhatóság jelentős előrelépést jelentenek a mobiltechnológia és az intelligens alkalmazások fejlődésében.

2.2.3. Big Transfer (BiT)

A Big Transfer (BiT) modellek olyan újító megközelítést jelentettek a vizuális tanulásában, amely jelentős javulást mutatott a különböző képfelismerési feladatokban. A "Big Transfer (BiT): General Visual Representation Learning" című cikk [Kolesnikov et al., 2020] bemutatja, hogy a nagy előtanított modellek finomhangolása különféle adathalmazokon hogyan vezethet javított teljesítményhez.

A BiT architektúra fő jellemzői:

- **Nagy méretű előtanítás:** a BiT modell előtanítása rendkívül nagy adathalmazokon történik, amely lehetővé teszi a modell számára, hogy általánosítható és robusztus vizuális anyagokon tanuljon.
- **Finomhangolás specifikus feladatokra:** A modellek általánosságát kihasználva ezeket különböző, specifikus vizuális felismerési feladatokra finomhangolják, ezzel maximalizálva a teljesítményt adott adathalmazokon;
- **Általánosíthatóság és robusztusság:** a BiT modellek kiemelkednek jól általánosítanak, ami lehetővé teszi széles körű alkalmazásokban való felhasználásukat, beleértve azokat a helyzeteket is, ahol az adatok jelentősen eltérnek az előtanításhoz használtaktól;
- **Csoport normalizáció és súly standardizáció:** a BiT modellek csoport normalizációt (Group Normalization, [Wu és He, 2018]), és súly standardizációt (Weight Standardization) használnak, amik javítják a nagy méretű adathalmazokon való tanulás hatékonyságát és elősegítik a jobb általánosítást;
- **Egyszerűsített finomhangolási protokoll:** a BiT-HyperRule heurisztika lehetővé teszi a finomhangolási hiperparaméterek egyszerű beállítását, csökkentve az új feladatokra való adaptálás költségeit.

[Kolesnikov et al., 2020] kutatása alapvető lépést jelent a gépi látás területén, bemutatva, hogy a nagyméretű előtanítás és a finomhangolás kombinációja kiváló eredményeket hozhat a vizuális feladatok széles körében. Ez a megközelítés új perspektívákat nyit a MI területén, elősegítve a hatékony és skálázható modellek fejlesztését.

2.3. Tudás desztilláció

A tudás desztilláció (Knowledge Distillation) egy olyan erőteljes technika a gépi tanulás, főként a mélytanulás területén amely 2015-ben jelent meg a [Hinton et al., 2015] cikk megjelenésével. A módszer a tudás átadására utal egy nagyobb modellből (tanár) egy egyszerűbb, kisebb modellbe (diák). A fő cél az, hogy lehetővé tegyünk a diák modell számára, hogy a lehető legközelebb teljesítsen a tanár modellhez, a tanár által rögzített kapcsolatok és mintázatok elsajátításával. Ez különösen hasznos olyan környezetekben, ahol a számítási erőforrások korlátozottak. A tudás desztilláció segít kiküszöbölni a teljesítmény és hatékonyság közötti kompromisszumot úgy, hogy a tudást egy kezelhetőbb formába sűríti jelentős teljesítménynyeresztés nélkül.

$$KL(p_t \parallel p_s) = \sum_{i \in \mathcal{C}} [-p_{t,i} \log p_{s,i} + p_{t,i} \log p_{t,i}] \quad (2.1)$$

A TD magába foglalja a tanár modell kimeneteinek felhasználását a tanítási fázisban, mint célokat a diák modellnek. Ez a KL divergencia segítségével történik (lásd 2.1 egyenlet), kiegészítve egy "hőmérséklet" (temperature) paraméterrel, ahol a $p_s \propto \exp\left(\frac{\log p_s}{T}\right)$ a tanár modell kimeneti valószínűségeire és a $p_t \propto \exp\left(\frac{\log p_t}{T}\right)$ pedig a diák modell kimeneteire utal. Mindez lehetővé teszi a diák modell számára, hogy megtanulja a tanár modell predikcióinak szerkezetét. A tudás desztilláció alkalmazásai változatosak, a modell kompressziótól a hatékony inferenciáig, valamint a korlátozott tanulási adatokkal rendelkező modellek tanulási folyamatának segítségéig terjedhetnek. Széleskörű alkalmazhatóságával ez továbbra is egy élénk kutatási terület, amely folyamatosan fejlődik új módszertanokkal, amelyek javítják a tudás átadásának hatékonyságát egyre hatékonyabb módon.

2.3.1. A jó tanár türelmes és következetes

A dolgozatunk a [Beyer et al., 2022] cikkekre alapszik amelyben a szerzők különböző tervezési döntéseket javasolnak, amelyek jelentősen befolyásolhatják a desztilláció hatékonyságát, és empirikus tanulmányokon keresztül alátámasztják az eredményeket.

Az felvetés miszerint a jó tanár konzisztens kell legyen abból áll, hogy a tanár és a diák modell ugyanazokat a kép nézeteket lássák a desztillációs folyamat során. Tehát azonos képvágást használtak mindkét modell számára, és a bevezették függvényillesztést (function matching), ami tovább bővíti ezt a megközelítést a mixup¹ adat augmentáció alkalmazásával.

A türelmes tanítási módszer jelentősége a függvényillesztésből ered. Akárcsak az alapvető tanulási folyamatok során, a desztillációs eljárás alatt is felvetődik a túl- és alulilleszkedés problémája: túl enyhe adataugmentáció esetén a modell hajlamos túlilleszkedni, míg túl intenzív adataugmentációnál alulilleszkedhet. Ha azonban a desztillációt mint függvényillesztést kezeljük, és biztosítjuk a konzisztens adatokat, akkor agresszíven alkalmazhatjuk az adataugmentációt is, miközben hatékonyak maradunk, feltéve, hogy elegendő ideig tanítjuk a diák modellt.

Adathalmazok terén a cikkek számos adatkészletet felhasználtak, mint például az ILSVRC-2012 (ImageNet) ([Deng et al., 2009]), Flowers102 ([Nilsback és Zisserman, 2008]), Pets ([Parkhi et al., 2012]), Food101 ([Jiang et al., 2020]) és a SUN397 ([Herranz et al., 2016]), ami alapján mi a kísérleteinket a Flowers102 adathalmazon végeztük.

Az elért eredmények különösen lenyűgözőek. Az ImageNeten például a standard ResNet-50 architektúrával, amit tudás desztillációval finomítottak, a tanulmány új csúcsteljesítményt ért el 82,8%-os top-1 pontossággal, amely jelentős javulást jelent a korábbi modellekhez képest. Ezen kívül a kisebb adathalmazokon, mint a Flowers102 és a Pets, szintén kiemelkedő javulásokat értek el.

2.3.2. További tudás desztilláció megközelítések

Koszinusz veszteség minimalizálása (Cosine Loss Minimization)

A koszinusz veszteség minimalizálásánál [Aguilar et al., 2020] a tanár modell belső reprezentációit igyekszünk a diák modellével hasonlítani. Ezt egy úgynevezett "CosineEmbeddingLoss"

¹A mixup egy olyan adat augmentációs technika, ami véletlenszerűen keveri össze a bemeneti képeket és a hozzájuk tartozó címkéket. Ezáltal a modellek jobban képesek kezelni az adatok közötti kis variációkat, és jobban általánosítanak az új adatokra.

2. FEJEZET: ELMÉLETI ALAPOK

segítségével tesszük, amely a belső állapotok közötti koszinusz hasonlóságot minimalizálja. Ez azt jelenti, hogy a diák modell tanulásának során nem csak a kimeneti rétegeket használjuk referencia pontként, hanem a rejtett rétegek vektoraiban is hasonlóságot próbálunk elérni. Ez azonban nem feltétlenül vezet jobb eredményre, mivel a két modell különböző architektúrával rendelkezik, és ezért nem biztos, hogy a tanár modell reprezentációja a legjobb lehetséges a diák számára. Az eltérő tanulási kapacitások miatt is problémás lehet egy ilyen közvetlen hasonlítás.

Közbenső regresszor (Intermediate Regressor)

A közbenső regresszor megközelítésnél [Romero et al., 2015] egy regresszor bevezetésével történik, ami a konvolúciós rétegek után kinyert jellemző térképeket hasonlítja össze. Itt a tanár és diák modell konvolúciós rétegei után kinyert jellemző térképek között próbálunk hasonlóságot találni, de egy tanítható regresszor segítségével, ami ideális esetben jobban összekapcsolja a két modell jellemzőit. A regresszor feladata, hogy segítse a dimenziók összehangolását, így egy hatékonyabb veszteségfüggvény definiálható a két modell között. Ez a veszteségfüggvény biztosítja a "tanítási utat", ami lehetővé teszi a gradiens visszaterjesztését, így módosítva a diák modell súlyait.

3. fejezet

Saját kísérletek

3.1. Használt architektúrák

Az elmúlt évek során a mélytanulási kutatásokban különös figyelmet kaptak az úgynevezett diák-tanár architektúrák. Ezek az architektúrák egy innovatív módszert kínálnak a tudás átadására egy nagyméretű, összetett tanár modellből egy kisebb, kevésbé bonyolult diák modellbe. Ezáltal a diák modell képes lesz a tanár modell által nyújtott tudás elsajátítására és annak viselkedésének imitálására, miközben jelentősen kevesebb számítási erőforrást igényel. Ebben a fejezetben részletezzük a kísérleti keretben használt, tanár szerepét betöltő BiT ResNet101 modellt, valamint a különböző diák modelleket, amelyeket a kutatás során alkalmaztunk.

3.1.1. Tanár

A [Kolesnikov et al., 2020] cikk keretében fejlesztett BiT ResNet101 modell, az [OxfordFlowers102](#) adathalmazra finomított változata szolgált a tanár modellként a kísérletekhez (a BiT modellek mélyreható ismertetése a 2.2.3 fejezetben található). Ez a modell azért volt alkalmas, mivel nyílt forráskódú, és a használt adathalmazon rendkívüli pontossággal bír (99%). Az előzetesen betanított modell súlyai nyilvánosan elérhetőek¹.

¹https://storage.googleapis.com/bit_models/vtab/BiT-M-R101x3-run2-oxford_flowers102.npz

3. FEJEZET: SAJÁT KÍSÉRLETEK

3.1. táblázat. Módosított ResNet18 architektúrák

layer name	output size	18-layer	18-layer / 2	18-layer / 4	18-layer / 8
conv1	112x112	7x7, 64, stride 2			
conv2_x	56x56	3x3 max pool, stride 2			
		$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 32 \\ 3 \times 3, 32 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 16 \\ 3 \times 3, 16 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 8 \\ 3 \times 3, 8 \end{bmatrix} \times 2$
conv3_x	28x28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 32 \\ 3 \times 3, 32 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 16 \\ 3 \times 3, 16 \end{bmatrix} \times 2$
conv4_x	14x14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 32 \\ 3 \times 3, 32 \end{bmatrix} \times 2$
conv5_x	7x7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$
	1x1	average pool, 1000-d fc, softmax			

3.1.2. Diák

MobileNetV3-small

Az első kísérlet során a MobileNetV3-small (lásd 2.2.2 fejezet) modellt választottuk diák modellnek, amely a [Howard et al., 2019] cikk alapján készült. Ez a modell ideálisnak bizonyult kezdeti kísérleteinkhez, mivel különösen jól optimalizálták mobil eszközök számára, ezzel biztosítva a kiváló sebességet és hatékonyságot még korlátozott számítási erőforrások mellett is. Az előre betanított változat lehetőséget adott arra, hogy a tanulási fázis gyorsabban történjen meg, anélkül, hogy az alapoktól kellett volna kezdenünk a folyamatot.

ResNet18

A további kísérleteink során a ResNet18 modellt és annak három módosított változatát használtuk, amelyeknél progresszíven csökkentettük a csatornák számát (lásd 2.2.1 fejezet, 3.1 táblázat). Az volt a célunk, hogy feltárjuk, hogyan hat a modell méretének és kapacitásának csökkentése a tudás átadásának hatékonyságára. Az első lépésben az alap ResNet18 modellt alkalmaztuk, majd lépésről lépésre csökkentettük a modell összetettségét, hogy megfigyelhessük a teljesítmény változásait.

3.2. Adathalmazok

A gépi tanulási kísérletek során a megfelelő adathalmaz kiválasztása kritikus fontosságú, mivel ez alapvetően befolyásolja a modellek tanulási képességét és teljesítményt. Ebben a fejezetben részletezzük az [OxfordFlowers102](#) és az [ImageNet1K](#) adathalmaz használatát, amely a kísérleteink alapját képezte. Az adathalmaz választásának okait és a kísérleti elrendezésben végrehajtott módosításokat is tárgyalni fogjuk.

3.2.1. ImageNet1K

Az előzetes tanítás fontos szerepet játszik a mélytanulási modellek hatékonyságában, különösen amikor specifikus feladatokra vagy adathalmazokra szeretnénk finomhangolni őket. Ebben a kontextusban a BiT ResNet101 modell, amely a tanár modellként szolgált a kísérleteimben, eredetileg az [ImageNet1K](#) adathalmazon lett előtanítva. Ez a széles körben használt, nagy méretű adathalmaz egy erős alapot biztosított a modellnek, lehetővé téve, hogy általánosíthatóbb jellemzőket tanuljon meg, mielőtt finomhangolásra került volna az [OxfordFlowers102](#) adathalmazra. Hasonlóan, a MobilNetV3-small modell, amit a kísérletek során diák modellként használtunk, szintén az [ImageNet1K](#) adathalmazon lett előre betanítva. Az előtanítás ebben az esetben is előnyös volt, mivel a modell már rendelkezett egy általános képfelismerési képességekkel, amelyeket aztán specifikusabb feladatokra, mint amilyen a virágfelismerés, tovább lehetett fejleszteni.

3.2.2. OxfordFlowers102

Az [OxfordFlowers102](#) egy széles körben használt adathalmaz a gépi tanulás és a képfeldolgozás területén, amely 102 különböző virágfajta képeit tartalmazza. Az adathalmaz jelentősége abban rejlik, hogy számos cikkben megjelenik, így az eredmények könnyen összehasonlíthatóvá válnak más kutatásokkal. Emellett méretét tekintve sem túl nagy, így nem igényel különösen nagy erőforrásokat a feldolgozáshoz, ami ideálissá teszi olyan kutatók számára, akik korlátozott számítási kapacitással rendelkeznek.

Az adathalmazt alapértelmezetten három részre van osztva: 1000 képből álló tanító és validációs halmaz, valamint egy 6000 képet tartalmazó teszt halmaz.

Kísérleteink során, szükségesnek találtuk a tanító és teszt adathalmazok felcserélését. Az in-

3. FEJEZET: SAJÁT KÍSÉRLETEK

doklás egyszerű: a ResNet modellek esetében nem álltak rendelkezésre előre betanított súlyok, így a modell teljesítménye jelentősen korlátozódott volna, ha csak a kisebb, 1000 adatpontot tartalmazó tanító halmazt használnom. A részek megcserélésével jelentősen megnövekedett a tanító adathalmaz mérete, ami lehetővé tette a modellek alaposabb és hatékonyabb tanítását. Ennek eredményeképpen jelentős pontosságnövekedést értünk el, amely reprezentatív eredmények elérését tette lehetővé.

Az OxfordFlowers102 adathalmaz használata és a kísérleti elrendezésben végrehajtott módosítások alapvetően hozzájárultak a kutatás sikeréhez. A tanító és teszt adathalmazok stratégiai felcserélése különösen fontos lépés volt a ResNet modellek hatékony tanítása szempontjából. Ez az adaptív megközelítés demonstrálja, hogy a gépi tanulási projektek során rugalmasnak kell lennünk az adathalmaz kezelésével és a kísérleti elrendezés optimalizálásával kapcsolatban, hogy a lehető legjobb eredményeket érjük el.

3.3. Tanítási módszerek

3.3.1. Tanulási stratégiák

Adatátalakítás

A tanítási adathalmazunk esetében a PyTorch transzformációs könyvtárát (transforms) használtuk az adatátalakítási lépések végrehajtásához. A transzformációk sorozata a következő műveletekből állt:

- **átméretezés:** az összes képet 256x256 pixel méretre méreteztük, ami egységes alapot biztosít a további transzformációkhoz;
- **véletlenszerű körülvágás:** az összes képet 224x224 pixel méretre hoztuk, ami egységes alapot biztosított a további transzformációkhoz;
- **véletlenszerű vízszintes tükrözés:** a képek véletlenszerű tükrözésével szimulálható a megjelenítés különböző orientációkban;
- **véletlenszerű forgatás:** A képeket maximum 30 fokkal forgattuk el, ami segít a modellnek jobban alkalmazkodni az orientációbeli változásokhoz;

3. FEJEZET: SAJÁT KÍSÉRLETEK

- **színállítás:** a fényerő, kontraszt, telítettség, és színezet véletlenszerű módosításával a modell jobban volt képes kezelni a különböző fényviszonyokat és színskálákat.

A képek végül tensor formátumba konvertálódtak, majd normalizálásra kerültek. A normalizálás során az átlagos RGB értékek $([0.485, 0.456, 0.406])$ és a szórások $([0.229, 0.224, 0.225])$ alapján skálázódtak a pixelek.

A validációs és tesztelési adathalmaz esetében egy kevésbé agresszív stratégiát alkalmaztunk, ami csupán az átméretezést, középre vágást, tensor konverziót, és normalizálást foglalta magába.

Tudás Desztilláció

A tudásátadási folyamat során a következő veszteségfüggvényt használtuk:

$$\text{Veszteség} = \left(\text{KL} \left(\frac{p_s}{T} \parallel \frac{p_t}{T} \right) \right) + H(y, p_s) \quad (3.1)$$

ahol KL a Kullback-Leibler divergencia (lásd 2.1 képlet), p_s és p_t a diák, illetve a tanár modellek kimenetei, T a hőmérsékleti paraméter, H a keresztentrópia veszteség (lásd 3.2 képlet), és y a valódi címkék.

$$H(y, p) = - \sum_i y_i \log(p_i) \quad (3.2)$$

Függvényillesztés

A [Beyer et al., 2022] cikket követve bizonyos kísérleteknél "mixup"-ot is alkalmaztunk, ami egyfajta függvényillesztési módszer, amelyről a [Zhang et al., 2017] cikk beszélt először. A mixup technika lényege, hogy a bemeneti képeket és azok címkéit véletlenszerűen keverjük össze, ezáltal új, szintetikus tanítási példákat létrehozva. Ez a megközelítés lehetővé teszi a modell számára, hogy robosztusabbá váljon az adatokban lévő zajokkal és eltérésekkel szemben, miközben segít a túlilleszkedés csökkentésében.

A függvényillesztési eljárás során két kép és azok címkéi között lineáris interpolációt végeztünk. A keverés mértéke egy λ paraméterrel szabályozható, amelyet bár alapjáraton a béta-eloszlásból sorsolunk ki, mi a cikket követve, egyenletes eloszlásból sorsoltunk, ezzel agresszívabbá téve a folyamatot. A tanítási példák így keletkezett kombinációja nemcsak, hogy

bővíti az adathalmazt, hanem a modellt is arra kényszeríti, hogy az interpolált címkék közötti átmeneteket is megértse, ezzel növelve a generalizáló képességeit.

3.3.2. Optimalizációs Algoritmusok

Optimalizációs algoritmusként **Adam**-et használtunk (lásd 2.1.5). A modell túlilleszkedésének elkerülése érdekében L2 regularizációt (más néven súlycsökkenést) is alkalmaztunk, az optimalizáció kiegészítéseként. Ez a módszer egy hozzáadott büntető tag a veszteségfüggvényben, ami az egyes súlyok négyzetes normájának arányában növeli a veszteséget. Az L2 regularizáció segít kontrollálni a modell komplexitását azzal, hogy bünteti a túl nagy értékű súlyokat, ezáltal elősegítve a simább és általánosíthatóbb funkciók megtanulását. Ezen felül, két különböző tanulási ráta ütemezőt is alkalmaztunk a tanulási folyamat optimalizálására: a **CosineAnnealingLR**-t és a **ReduceLROnPlateau**-t.

A **CosineAnnealingLR** ütemező a tanulási rátát a tanítási ciklusok során periodikusan csökkenti, egy előre meghatározott minimumig, követve ezzel a koszinusz függvény alakját. Másrészt, a **ReduceLROnPlateau** ütemező a tanulási rátát csökkenti, ha a validációs veszteség nem javul egy meghatározott számú epoch után.

A tanítási folyamat további optimalizálása érdekében korai leállást (early stopping) is implementáltunk. Ez a technika abbahagyja a tanítást, ha a validációs teljesítmény egy adott számú epochán keresztül nem mutat jelentős javulást, ezzel megelőzve a modell túlilleszkedését és időtakarékosan kezelve az erőforrásokat.

3.3.3. Értékelési metrikák és validálás

A mélytanulási modellek értékelése során a teljesítmény objektív méréséhez számos metrikát alkalmaztunk. Ezek a metrikák lehetővé teszik, hogy pontos képet kapjunk a modell viselkedéséről, képességeiről és a tanítási folyamat során bekövetkezett fejlődésről.

Pontosság és Veszteség

A modell pontossága (top1 accuracy) az egyik leggyakrabban használt metrika, amely azt méri, hogy a modell milyen arányban adott helyes előrejelzéseket a tesztelés során. A pontosság kiszámítása egyszerű: az összes helyesen osztályozott eset számát elosztjuk az összes teszt eset

3. FEJEZET: SAJÁT KÍSÉRLETEK

számával. Ezt a metrikát különösen fontos figyelemmel kísérni, mivel közvetlen információt szolgáltat a modell általánosító képességéről.

A veszteség (loss) mértéke szintén kritikus jelentőségű, hiszen ez mutatja meg, mennyire tér el a modell előrejelzése a valós címkéktől. A **keresztentropia** veszteségfüggvényt használtuk, amely a kategóriák közötti távolságot számítja ki, és a modell által adott valószínűségek és a tényleges címkék közötti eltérést méri. Ez a metrika segít felmérni, hogy a modell mennyire "biztos" az egyes előrejelzésekben, és jelentős mértékben hozzájárul a hibák diagnosztizálásához és a modell finomhangolásához.

3.3.4. Kísérleti környezet

Az oktatási folyamat során a Google Colab platformjának PRO verziójával dolgoztunk, amely többféle számítási erőforrást foglal magába, mint például T4, A100, L4, és V100 típusú GPU-kat, valamint Tensor Processing Unit (TPU) támogatást is biztosít. Ezen kívül a Kaggle felületét is igénybe vettük, amely további két T4-es GPU-val járult hozzá a projekthez. A tanulási folyamat monitorozásához a Weights and Biases platformot alkalmaztuk

A Google Colab ingyenes változatának korlátozott számítási kapacitása jelentős akadályt jelentett a projektünk számára. Bár a platform biztosított egy NVIDIA Tesla T4 GPU-t, ennek használati ideje hetente korlátozott volt, ami nem tette lehetővé folyamatos számítási igényeink kiszolgálását. Alternatívaként a Kaggle platformot vetettük be, amely lehetővé tette heti harminc óra számítási idő használatát két T4 GPU-val, de ez a megoldás számos más szempontból nem bizonyult ideálisnak. Ezen GPU-k sebessége nem volt elegendő az általunk használt adathalmazok és modellek számára. Továbbá, ellentétben a Colab-bal, a Kaggle nem nyújtott natív támogatást a Google Drive szolgáltatásnak, ami kulcsfontosságú volt modelljeink súlyainak automatikus mentésében és betöltésében. Ezen felül a Kaggle felhasználói felülete is alacsonyabb minőséget nyújtott, mint a Colabé.

3.4. Eredmények

A kutatás során végzett kísérletek célja az volt, hogy feltárjuk, miként lehet a nagy, összetett modellekből származó tudást hatékonyan átadni kisebb, egyszerűbb szerkezetű hálózatokba. A vizsgált modellek között szerepelnek a MobileNet és a ResNet18 különböző változatai,

3. FEJEZET: SAJÁT KÍSÉRLETEK

	MobileNet	MobileNet Distilled	MobileNet Distilled FunMatch
Batch méret	32	32	32
Epochok	55	37	40
Tanulási ráta	1.00E-03	1.00E-03	1.00E-04
Súlycsökkenés	1.00E-06	1.00E-06	1.00E-08
Hőfok	N/A	1	1

3.2. táblázat. MobileNet konfigurációk

	MobileNet	MobileNet Distilled	MobileNet Distilled FunMatch
Paraméterszám	1,622,406		
Validációs hiba	0.1294	0.1673	0.3483
Pontosság	0.9627	0.9804	0.9667

3.3. táblázat. MobileNet eredmények

amelyeket csatornaszám-csökkentéssel módosítottunk. Az eredmények bemutatása során külön figyelmet fordítunk a modellméretek és a tanulási teljesítmény kapcsolatának elemzésére, valamint a desztilláció során alkalmazott különböző technikák összehasonlítására. Az eredmények számszerűsítésével és a kísérletek metodikus leírásával kívánjuk biztosítani a vizsgálatok reprodukálhatóságát és értékelhetőségét.

3.4.1. Konfigurációk és kvantitatív eredmények

A kísérleteink során a ResNet18 architektúra különböző változataival és a MobileNet architektúrákkal foglalkoztunk. Minden modell esetében pontosan dokumentáltuk a konfigurációkat illetve az eredményeket, amelyek a 3.4, 3.5 és a 3.2, 3.3 táblázatokban találhatóak meg. A ResNet modelleknél minden esetben véletlenszerűen inicializált súlyokkal kezdtük a tanulási folyamatot, mivel egyik modell sem volt előtanítva. Ezzel szemben a MobileNet modellt előtanított súlyokkal használtuk, ami jelentősen felgyorsította a tanulási és validációs veszteség konvergenciáját, valamint növelte a modell precizitását. A desztillációs folyamat során a konvergencia lassúsága miatt több iterációt kellett futtatnunk, mint a hagyományos tanítás esetén. A függvényillesztési technika tesztelése során pedig csökkentenünk kellett a tanulási rátát és a súlycsökkentést, mivel a módszer maga erőteljes regularizációs hatással bír, emellett szükséges volt az epochok számának további növelésére is, ahogyan az várható is volt a [Beyer et al., 2022] cikk alapján. Mindemellett a nyolcadolt csatornaszámú Resnet esetén gradiens vágást

3. FEJEZET: SAJÁT KÍSÉRLETEK

3.4. táblázat. ResNet18 változatainak konfigurációi, ahol D: desztillált változat, F: függvényillesztés volt alkalmazva, R18_x: x-szer leméretezett modell

	R18	R18 D.	R18 D. F.	R18_2	R18_2 D.	R18_4	R18_4 D.	R18_8	R18_8 D.
Batch méret	32	32	64	32	32	32	32	32	32
Epochok	60	69	80	49	59	59	69	69	69
Tanulási ráta	1E-03	1E-03	1E-04	1E-03	1E-03	1E-03	1E-03	1E-03	1E-03
Súlycsökkenés	1E-05	1E-05	1E-08	1E-05	1E-05	1E-05	1E-05	1E-05	1E-05
Hőfok	N/A	1	1	N/A	1	N/A	1	N/A	1

3.5. táblázat. ResNet18 változatainak eredményei ahol D: desztillált változat, F: függvényillesztés volt alkalmazva, R18_x: x-szer leméretezett modell

Méret és típus	R18	R18 D.	R18 D. F.	R18_2	R18_2 D.	R18_4	R18_4 D.	R18_8	R18_8 D.
Paraméterszám	12,609,894			3,171,558		802,470		205,446	
Validációs hiba	0.458	0.3326	0.6303	0.4698	0.387	0.6738	0.4329	0.7696	0.7702
Pontosság	0.8735	0.9431	0.898	0.87	0.9314	0.8098	0.9039	0.7696	0.7961

kellett alkalmaznunk, modell méretéből adódó gradiens robbanás probléma megoldásáért.

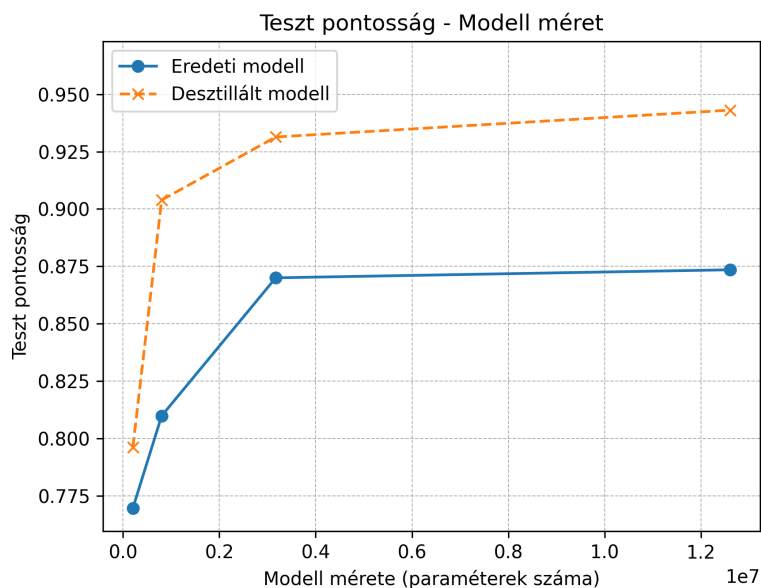
3.4.2. Eredmények elemzése

Az adatok elemzése egyértelműen kimutatta, mint ahogy a 3.1 ábra is mutatja, hogy a tudás desztillációs módszer jelentős mértékben javítja a modell teszt pontosságát. Specifikusan, a ResNet18 architektúra standard és felezett csatornaszámmal rendelkező változatai között megfigyelhető volt egy 6-7%-os teljesítménynövekedés. A legnagyobb javulást a negyedelt csatornaszámmal rendelkező változat érte el, ahol a pontosság akár 9%-kal is növekedett. Ezzel szemben, az nyolcadolt csatornaszámmal rendelkező változatnál már csak 3%-os javulás volt tapasztalható, ami valószínűleg a modell túlzottan kis méretének tulajdonítható.

Az a 3.3 táblázat alapján a MobileNettel végzett kísérletek során az észlelt 2%-os pontosságjavulás, bár abszolút értékben csekélynek tűnhet, jelentősegteljesnek minősül. Ennek oka, hogy a tanár modell és a diák modell közötti eredeti pontosságbeli különbség csupán 3%. Így a desztillált diákmodell pontossága már csak 1%-kal marad el a tanármodelltől. Ezen eredmény elérésében kulcsfontosságú szerepet játszott a MobileNet ImageNet-en előtanított verziójának felhasználása.

Továbbá, ahogy azt a [Beyer et al., 2022] cikk ajánlása alapján tettük, teszteltük a függvény-

3. FEJEZET: SAJÁT KÍSÉRLETEK

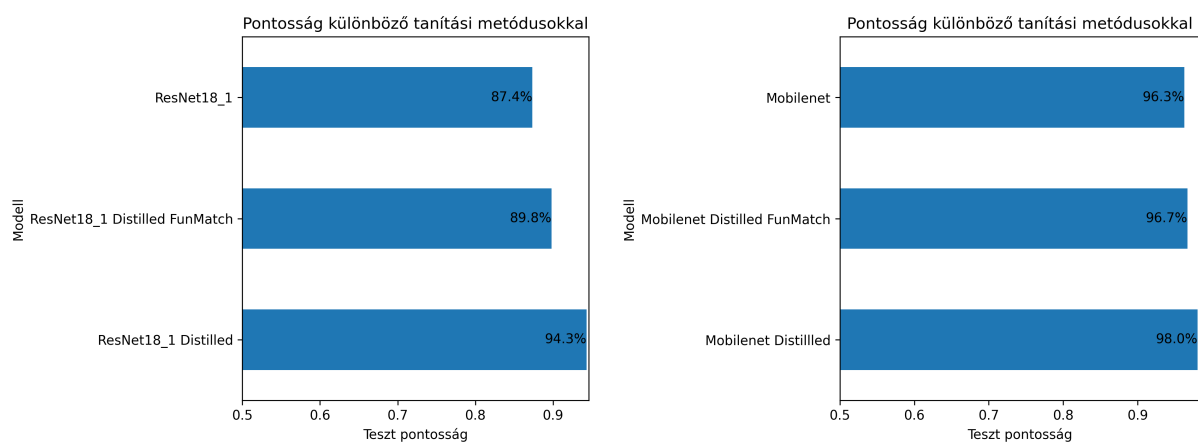


3.1. ábra. Eredeti és desztillált modellek tesztpontosságának összehasonlítása a modell méretének növekedésével. A grafikon azon kapcsolatot ábrázolja, hogy a modell mérete (a paraméterek száma szerint) hogyan befolyásolja a tesztpontosságot mind az eredeti modell (kék színnel és szilárd vonallal), mind a desztillált modell (narancssárga színnel és szaggatott vonallal) esetében.

illesztési, más néven adatkeverési (mixup) technikát. Az a 3.2 és a 3.2 ábrák szerint sajnos ez a módszer nem hozta meg a várt eredményeket az esetünkben, mivel nem javította a pontosságot az egyszerű desztillációhoz képest. Viszont, ami előre látható volt, a tanulási veszteség jelentősen fluktuált, és a konvergáció is később következett be. Egy lehetséges magyarázat arra, hogy miért nem volt sikeres a módszer a mi esetünkben, az lehet, hogy a tanítások túl rövidek voltak ahhoz, hogy a technika kifejthesse a hatását.

3. FEJEZET: SAJÁT KÍSÉRLETEK

3.2. ábra. Pontosság összehasonlítása különböző oktatási módszerekkel Resnet18 és Mobilenet-V3-Small modellekre.



4. fejezet

Konklúzió

Eredményeink egyértelműen kimutatták, hogy a tudás desztilláció jelentős mértékben javította a diák modellek teljesítményét, különösen a ResNet18 architektúra változatainál. A legnagyobb pontosság növekedést a negyedelt csatornaszámmal rendelkező változat mutatta, ahol a teszt pontosság akár 9%-kal is növekedett. Bár a nyolcadolt csatornaszám esetében a javulás mérsékeltebb volt, ez a modell kis mérete miatt várható volt.

A MobileNet esetében a desztilláció 2%-os pontosságjavulást eredményezett, ami figyelemre méltó, tekintettel arra, hogy az eredeti pontossági különbség a tanár és a diák modell között mindössze 3% volt. Az ImageNet-en előtanított MobileNet modell használata kritikus szerepet játszott ebben az eredményességben, mivel a diák modell már rendelkezett általános képfelismerési képességekkel.

A függvényillesztési (mixup) technika alkalmazása nem hozta meg a várt előnyöket, mivel nem javította a pontosságot az egyszerű desztillációhoz képest, és a tanulási veszteség fluktuációt mutatott. Ez arra utal, hogy bár ez a módszer növelheti a modell robusztusságát, nem minden esetben előnyös a pontosság szempontjából.

Összességében a kutatásunk alátámasztotta, hogy a diák-tanár architektúrák alkalmazása jelentősen javíthatja a kisebb modellek teljesítményét, miközben csökkenti a számítási és erőforrás-igényeket. Ezek az eredmények hozzájárulnak a gépi tanulási modellek hatékonyabb és elérhetőbb alkalmazásához különböző gyakorlati környezetekben.

4.1. További kutatási lehetőségek

A jelenlegi kutatás során feltártuk a diák-tanár architektúrák előnyeit a képfeldolgozás terén, azonban több ígéretes irány is felmerül, amelyek tovább mélyíthetik tudásunkat és növelhetik a

4. FEJEZET: KONKLÚZIÓ

modellalkalmazások hatékonyságát. Az alábbiakban vázoljuk a további kutatási lehetőségeket:

- **Transformer alapú modellek integrálása:** Az utóbbi években a transformer architektúrák, mint például a Vision Transformer (ViT), jelentős előrelépést hoztak a képfeldolgozás terén. Ezen modellek különösen alkalmasak lehetnek arra, hogy diák modellként szolgáljanak a diák-tanár paradigmában, mivel képesek nagyobb kontextusban értelmezni a vizuális információkat.
- **Mélyebb tudásátadási mechanizmusok:** További kutatásokra van szükség a tudásátadási folyamatok mélyebb megértése érdekében. Például, a tudás desztillációban alkalmazott hőmérsékleti paraméter finomhangolása vagy új veszteségfüggvények, mint például a kontrasztív¹ vagy triplet² veszteség, bevezetése jelentősen javíthatja a tanulási eredményeket.
- **Különböző adathalmazokon való tesztelés:** A jelen kutatás az OxfordFlowers102 és ImageNet adathalmazokra korlátozódott. A jövőbeli kutatásokban érdemes lenne más, változatosabb adathalmazokon is tesztelni a modelleket, beleértve a ritkábban használt vagy újonnan létrehozott adathalmazokat, amelyek kihívást jelenthetnek a modellek számára.

¹A kontrasztív veszteség (contrastive loss) olyan veszteségfüggvény, amit páros adatpontokhoz alkalmaznak a mélytanulási modellekben. Célja, hogy növelje a különböző kategóriák mintái közötti távolságot, miközben csökkenti az azonos kategóriák mintái közöttit. Alkalmazása ideális olyan feladatokra, mint az arcfelismerés vagy objektum azonosítás, ahol fontos a minták közötti finom különbségek megragadása.

²A triplet veszteség (triplet loss) három adatpontot használ (alap, pozitív, negatív) a tanítás során. A modell célja, hogy növelje az alap és a negatív minta közötti távolságot, miközben minimalizálja az alap és a pozitív minta közöttit. Ez a veszteségfajta előnyös olyan alkalmazásokban, ahol a relatív különbségek kiemelkedően fontosak, mint például képfelismerési feladatokban vagy beágyazott reprezentációk tanításában.

5. fejezet

Köszönetnyilvánítás

Ezúton szeretném kifejezni köszönetemet Pável Szabolcsnak, témavezetőmnek, aki türelmével és útmutatásaival végig támogatta kutatási munkámat. Külön köszönet illeti Máthé Gergelyt és Poór Veronikát, akik a hozzájárultak a kutatás kiinduló ötletéhez és segítettek eligazodni a témakeresésben.

Köszönöm mindannyiuknak, hogy hozzájárultak tudományos munkám elkészültéhez és lehetővé tették, hogy feltérképezzem kutatási témám lehetőségeit.

Irodalomjegyzék

- Aguilar, G., Ling, Y., Zhang, Y., Yao, B., Fan, X., és Guo, C. Knowledge distillation from internal representations. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7350–7357, 2020.
- Beyer, L., Zhai, X., Royer, A., Markeeva, L., Anil, R., és Kolesnikov, A. Knowledge distillation: A good teacher is patient and consistent. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10925–10934, 2022.
- Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., és others, . Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., és Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.
- Floridi, L. és Chiriatti, M. Gpt-3: Its nature, scope, limits, and consequences. *Minds and Machines*, 30:681–694, 2020.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., és Bengio, Y. Generative adversarial networks. *Communications of the ACM*, 63(11): 139–144, 2020.
- He, K., Zhang, X., Ren, S., és Sun, J. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778. IEEE, 2016.
- Hebb, D. *The Organization of Behavior*. Wiley & Sons, New York, 1949.
- Herranz, L., Jiang, S., és Li, X. Scene recognition with cnns: objects, scales and dataset bias. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 571–579, 2016.
- Hinton, G., Vinyals, O., és Dean, J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.
- Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., és others, . Searching for mobilenetv3. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019.
- Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., és Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.
- Ioffe, S. és Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015. URL <http://arxiv.org/abs/1502.03167>.
- Jiang, L., Qiu, B., Liu, X., Huang, C., és Lin, K. Deepfood: food image analysis and dietary assessment via deep model. *IEEE Access*, 8:47477–47489, 2020.

- Kingma, D. és Ba, J. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 12 2014.
- Kolesnikov, A., Beyer, L., Zhai, X., Puigcerver, J., Yung, J., Gelly, S., és Houlsby, N. Big transfer (bit): General visual representation learning. *arXiv preprint arXiv:1912.11370*, 2020. URL https://github.com/google-research/big_transfer.
- Krizhevsky, A., Sutskever, I., és Hinton, G. E. Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C., Bottou, L., és Weinberger, K., editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. URL https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.
- LeCun, Y., Bottou, L., Bengio, Y., és Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- Li, C.-X., Fei, W.-M., Shen, C.-B., Wang, Z.-Y., Jing, Y., Meng, R.-S., és Cui, Y. Diagnostic capacity of skin tumor artificial intelligence-assisted decision-making software in real-world clinical settings. *Chinese medical journal*, 133(17):2020–2026, 2020.
- Nilsback, M.-E. és Zisserman, A. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, pages 722–729. IEEE, 2008.
- Parkhi, O. M., Vedaldi, A., Zisserman, A., és Jawahar, C. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE, 2012.
- Pitts, W. és McCulloch, W. A logical calculus of the ideas immanent in nervous activity. *Journal of Mathematical Biophysics*, 5:115–133, 1943.
- Robbins, H. és Monro, S. A Stochastic Approximation Method. *The Annals of Mathematical Statistics*, 22(3):400 – 407, 1951. doi: 10.1214/aoms/1177729586. URL <https://doi.org/10.1214/aoms/1177729586>.
- Romero, A., Ballas, N., Kahou, S. E., Chassang, A., Gatta, C., és Bengio, Y. Fitnets: Hints for thin deep nets, 2015.
- Rosenblatt, F. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.
- Rumelhart, D. E., Hinton, G. E., és Williams, R. J. Learning representations by back-propagating errors. In *Nature*, volume 323, pages 533–536, 1986.
- Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., és Chen, L.-C. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- Simonyan, K. és Zisserman, A. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., és Rabinovich, A. Going deeper with convolutions. In *IEEE conference on computer vision and pattern recognition*, pages 1–9. IEEE, 2015.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., és Polosukhin, I. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017.
- Wang, F.-Y., Zhang, J. J., Zheng, X., Wang, X., Yuan, Y., Dai, X., Zhang, J., és Yang, L. Where does alphago go: From church-turing thesis to alphago thesis and beyond. *IEEE/CAA Journal*

IRODALOMJEGYZÉK

of Automatica Sinica, 3(2):113–120, 2016.

Wu, Y. és He, K. Group normalization. *CoRR*, abs/1803.08494, 2018. URL <http://arxiv.org/abs/1803.08494>.

Zhang, H., Cisse, M., Dauphin, Y. N., és Lopez-Paz, D. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

A. függelék

Felhasznált eszközök

A dolgozat elkészítése során az OpenAI ChatGPT ([Brown et al., 2020]) mesterséges intelligencia modelljét használtam nyelvtani és stilisztikai hibák azonosítására és javítására. Minden javaslatot alapos emberi felülvizsgálat követett annak érdekében, hogy a szöveg megőrizze eredeti szándékamat és tudományos pontosságát.