Budapesti Műszaki és Gazdaségtudományi Egyetem
Villamosmérnöki és Informatikai Kar
BSc Mérnökinformatika Szakirány [CE]
Távközlési és Mesterséges Intelligencia Tanszék
Információs Rendszerek Specializáció
Adatelemzés Mélytanulási Módszerekkel Ágazati Főtantárgy

FungiCLEF: Gombák Klasszifikációja Multi-head Konvolúciós Hálókkal és Finomhangolt MetaFormerrel

Czimber Márk
szerző

Kocsis Dávid
szerző

Márkó Ágnes
szerző

Dr. Papp Dávid
témavezető
egyetemi adjunktus

Dr. Mihajlik Péter
oktató
egyetemi adjunktus

Dr. Szűcs Gábor
oktató
egyetemi docens

2024, Szorgalmi Nagyházi Feladat

1. Bevezetés

A FungiCLEF 2024 a LifeCLEF kihívás részeként valósul meg, amely a szélesebb CLEF kezdeményezés keretében működik, és célja a gépi tanulás fejlesztése a biodiverzitás kutatásának támogatására. A FungiCLEF verseny fókuszában a gombafajok automatikus azonosítása áll fényképek és metaadatok elemzése révén. A kezdeményezés kiemelt célja, hogy támogassa a mikológusokat, amatőr természetkutatókat és természetkedvelőket a vadon élő gombák meghatározásában, ami különösen fontos a biodiverzitás dokumentálása és az ökológiai kutatások szempontjából. A gombák létfontosságú szerepet játszanak a tápanyagok újrahasznosításában és a növényekkel való szimbiotikus kapcsolatok fenntartásában, ami kiemeli jelentőségüket az ökoszisztémákban [1]. Noha a gombák az élővilág egyik legváltozatosabb és ökológiailag kiemelkedő jelentőségű csoportját alkotják, azonosításuk számos nehézségbe ütközik, például a fajok közötti vizuális hasonlóság, a környezeti hatások és a terepi szakértők korlátozott hozzáférhetősége miatt. Az egyértelmű osztályozás a közegészségügy szempontjából is kritikus, mivel több gombafaj erősen mérgező. A FungiCLEF feladat egy olyan platformot kínál, amely lehetővé teszi olyan gépi tanulási modellek fejlesztését, amelyek integrálják a vizuális információkat és a metaadatokat, ezáltal javítva az azonosítás pontosságát.

1.1. Motiváció

A legfőbb motivációnk a feladathoz abból ered, hogy a képek gyakran több információt hordoznak számunkra, mint az azonos idő alatt feldolgozható szöveg. Napi szinten számtalan információt dolgozunk fel a környezetünkből pusztán a látvány alapján; bár olvasás útján is sok információhoz jutunk, ez nem éri el a vizuális észlelés mértékét. Ezt a jelenséget a "kép-fölény hatás" (picture superiority effect) néven ismerjük, amely szerint a képek könnyebben megjegyezhetők és visszahívhatók, mint a szavak. Ez a hatás különösen erős, amikor a képek és szavak együttesen jelennek meg, mivel a vizuális információk erőteljesebben rögzülnek a memóriában. Továbbá, a vizuális észlelés gyorsabb és párhuzamos feldolgozást tesz lehetővé, ami hozzájárul a hatékonyabb információfeldolgozáshoz. A csecsemők sokáig csak a jelenet megfigyeléséből tanulnak, képsorozatok megfigyeléséből értik meg a környezetet, melyet később megerősítenek a szöveg hozzácsatolásával. Ez a gondolkodás átvezethető mélytanulás alapú megoldásokra [11], alátámasztva a vizuális adatok alkalmazásának előnyeit a gépi tanulási feladatokban és a feladat pontosságának növelését szöveg alapú leírás hozzáadásával. Szerencsére a multimodális nagy nyelvi modellek képesek mindkét modalitás egyidejű feldolgozására, és erősebb vizuális megértési képességet mutatnak párhuzamos használatukkor [8]. Ez a lehetőség rendkívül előnyös, hiszen a látás és a nyelv az emberi intelligencia két alapvető képessége. A FungiCLEF pontosan ebbe a kategóriába tartozik, ahol a modellnek képesnek kell lennie pusztán vizuális információ alapján meghatározni egy gombafajt, valamint a név és a kép alapján eldönteni annak mérgező jellegét. A pontos azonosítás elősegíti a terepmunka során vagy a közfelhasználás szempontjából a fogyasztható gombák meghatározását és a mérgező fajok elkerülését, így pozitív hatással van mind az ökoszisztémára, mind a társadalomra. Ez a közös cél vezérelt minket abban, hogy részt vegyünk a FungiCLEF kihívásban, hozzájárulva a meglévő megoldásokhoz és lehetőség szerint továbbfejlesztve azokat.

1.2. Nehézségek

Zajos és heterogén adatok kezelése: A képfeldolgozás során előforduló zajok, mint például az érzékelői zaj vagy a kvantálási zaj, torzíthatják a képeket, így csökkentve a modellek teljesítményét. [13]. Apró és nehezen észlelhető vizuális különbségek: A gombafajok közötti morfológiai hasonlóságok miatt a vizuális különbségek gyakran minimálisak és nehezen észlelhetők, a gombák változatos megjelenése és a fajok közötti finom különbségek tovább nehezítik a pontos azonosítást. [5]. Kis méretű adathalmazok és túltanulás: A korlátozott méretű adathalmazok miatt a gépi tanulási modellek hajlamosak a túltanulásra, ami rontja a generalizációs képességüket új adatokon. Kiegyensúlyozatlan adathalmazok: Az adathalmazok gyakran

kiegyensúlyozatlanok a gombafajták szerinti képmennyiségek tekintetében, különösen a mérgező gombák esetében, amelyekből jóval kevesebb kép áll rendelkezésre. Ez az egyensúlyhiány megnehezíti a modellek számára a mérgező gombák felismerését, mivel kevesebb példát látnak ezekből az osztályokból tanuláskor. Végül a metaadtokban rejlő lehetőségek nem kihasználhatóak, hiszen az adathalmaz Dániára van korlátozva.

1.3. Tématerület

A gombák ökológiai és gazdasági szempontból is jelentősek. Több millió ismert fajjal a gombák sokfélesége kiemelkedő. A pontos taxonómiai besorolás lehetővé teszi a gombák ökológiai funkcióinak és interakcióinak mélyebb megértését, ami elengedhetetlen az ökoszisztémák egészségének fenntartásához. A szakirodalom számos módszert alkalmaz a gombafajok osztályozására, beleértve a morfológiai jellemzők elemzését és a molekuláris technikákat, mint például a DNS-szekvenálást. A hagyományos osztályozás morfológiai alapú és gyakran szubjektív [10]. Gépi tanulási módszerekkel hatékonyabb és skálázhatóbb megoldások érhetők el. Egy olyan folyamat, amely korábban órákig tartó irodalomkutatást igényelt, ma már másodpercek kérdése. Például a Dán Gombák Atlasza által nyújtott gombaazonosító szolgáltatás segítségével a felhasználók egyszerűen készítenek egy fényképet a megfigyelésükről, és a rendszer azonnal javaslatot tesz a lehetséges fajokra [4]. Ez lehetővé teszi a felhasználó számára, hogy manuálisan ellenőrizze a javaslatot, összehasonlítva a megfigyelést a fajok fényképeivel és a hozzájuk tartozó leírásokkal [7]. Ez a lehetőség fogalmazta meg a FungiCLEF gombaosztályozási kihívást, mely mikológusok, kutatók és lelkes amatőrök munkáját segíti a gombafajok hatékony azonosításában. A probléma megközelítése mélytanulási szempontból két felé ágazik, előre betanított prior eloszlással rendelkező látásmodellek használata, mint az EfficientNet [9], a VisionTransformer [3] és a MetaFormer [14]. Ezek a modellek sokszor hosszú betanítási idővel és jelentős memóriaigény használattal rendelkeznek. A másik megközelítés egy saját, esetlegesen kisebb mélytanulási architektúra definiálása, mely kevesebb paraméterrel hasonlóan meg tudja tanulni az adathalmaz eloszlását, mindezt kevesebb számítási kapacitással rendelkező számítógépen. A rendelkezésre álló erőforrások és megközelítési módszerek sokasága miatt, a probléma megfogalmazása inkább egy tervezési döntés kérdésére.

1.4. Megközelítés

A kutatásunk során a sokszínűség és a modularitás elveire helyeztük a hangsúlyt, hogy egy skálázható, rugalmasan fejleszthető és bővíthető rendszerarchitektúrát alakítsunk ki. E cél elérésére több, a szakirodalomban megalapozott és modern megközelítést alkalmaztunk. Négy különböző modellarchitektúrát választottunk, amelyek a jelenlegi legkorszerűbb módszerekre és technológiákra épülnek:
1. Saját fejlesztésű Multi-Head CNN modell: Ez az architektúra teljesen az alapoktól került felépítésre, lehetőséget adva a teljes rendszer irányítására és optimalizálására.
2. ResNet50 alapú AdaBoost és Gradient Boosting klasszifikátorok: Ez az architektúra egy robusztus, mély tanulási alapmodellt kombinál klasszikus, erősítő tanulási technikákkal.
3. Finomhangolt EfficientNet-b0 Hybrid Multi-Head CNN: Az EfficientNet-b0 modellt továbbfejlesztve és hibridizálva egy Multi-Head CNN struktúrával.
4. Finomhangolt Multi-Task Metaformer alapú Casual Transformer megközelítés: Ez az innovatív architektúra az egyik legmodernebb Metaformer struktúrát használja és a transformer-alapú rendszerek rugalmasságát és skálázhatóságát ötvözi konvolúciós hálókkal.
A finomhangolás lehetővé tette a modellek specifikus alkalmazkodását az adathalmazhoz. Megalkotásukkor a szakirodalomban elismert építőkövekre támaszkodtunk, miközben megvalósításuk során különös figyelmet kapott a moduláris tervezés és a cserélhetőség biztosítása. A modellek multi-head klasszifikációra történő optimalizálása lehetővé teszi a gombafajok széles spektrumának és azok mérgezettségének egyidejű figyelembevételét. A feldolgozott adathalmaz, amely 100 különböző gombafajt tartalmaz, a PyTorch adatbetöltő mechanizmusán keresztül kerül a modellekhez. A kutatás szerves részét képezi egy részletes ablációs vizsgálat, amely során összehasonlítjuk az egyes architektúrák előnyeit és hátrányait.

2. Gomba-klasszifikáció

A gombák osztályozása a képosztályozás egyik különleges területe, mivel a gombák rendkívül változatos és összetett biológiai tulajdonságokkal rendelkeznek. A gombafajok között vizuális hasonlóság figyelhető meg, ami különösen megnehezíti az osztályozást, hiszen a mérgező és ehető fajok megkülönböztetése néha csak apró morfológiai jegyek alapján lehetséges. Emellett a környezeti tényezők, például az élőhely és az évszakok, jelentősen befolyásolják a gombák megjelenését. A gépi tanulás, különösen a konvolúciós neurális hálózatok alkalmazása, lehetőséget nyújt ezen komplexitás kezelésére, ugyanakkor az adatok egyensúlytalansága, például az alulreprezentált fajok problémája, kihívást jelent. Az ilyen projektek nemcsak a gombák biodiverzitásának jobb dokumentálását segítik elő, hanem a terepi alkalmazások, például mérgezések megelőzésére szolgáló eszközök fejlesztésében is fontos szerepet játszanak. Összességében a gombák osztályozása olyan kihívás, amely a gépi tanulás és az ökológia határterületeit egyesíti, hozzájárulva a tudományos és gyakorlati előrehaladáshoz.

2.1. Adathalmaz

A FungiCLEF 2024, a LifeCLEF keretében megrendezett gombaazonosító kihívás második kiadása, a Danish Fungi 2020 adatbázis és az előző verseny tapasztalataira épül, új feladatokkal bővítve a kihívást. Ezek a feladatok különböző gyakorlati forgatókönyveket tükröznek, például a mérgező és ehető fajok megkülönböztetését vagy eddig nem azonosított fajok felfedezését [7]. Az adatbázis több mint 295,000 fényképet tartalmaz 1,604 gombafajról, amelyeket főként Dániában dokumentáltak, és amelyek különböző élőhelyeket és évszakokat reprezentálnak. Ez a sokféleség ideális környezetet teremt gépi tanulási modellek számára. Az adatok hitelességét szakértők által validált annotációk garantálják. A vizuális tartalmak mellett metaadatok is rendelkezésre állnak, amelyek tartalmazzák az élőhely típusát, a szubsztrátumot, a földrajzi helyet, a megfigyelés időpontját és az EXIF-adatokat. Ezek a kiegészítő információk lehetővé teszik, hogy a modellek nemcsak vizuális, hanem ökológiai kontextust is felhasználjanak a fajok azonosításához. A mi specifikus feladatunk egy olyan osztályozó modell fejlesztése volt, amely az eredeti adatbázisból kiválasztott 100 faj pontos azonosítására képes. A fajok kiválasztása során figyelembe vettük az egyes fajokhoz tartozó tanító adathalmaz méretét, vagyis a fajokról készített képek számát, azok méretét, valamint a mérgező és ehető fajok arányát a kiválasztottak esetén. A mi esetünkben ez az arány 10:90 lett, ahol a mérgező fajok reprezentálják a kisebb csoportot (az eredeti adathalmazban valamivel kisebb ez az arány). A kiválasztott csoportok 180 és 320 közötti számban tartalmaznak képeket, átlagosan 260 képpel, amelyek mind 300 pixel szélességb, magasságuk pedig 200-300 pixel között változik. Az adatfeldolgozás kulcsszerepet játszik az ilyen típusú kihívások során. Az adatok előkészítése során gyakori lépések közé tartozik a képnormalizáció, amely során a pixelértékeket egy egységes skálára állítjuk (például 0–1 közé), és az adatok augmentálása, amely mesterségesen növeli az adathalmaz sokféleségét forgatással, eltolással vagy tükrözéssel. Ez különösen fontos a gombák vizuális hasonlósága miatt, amely nehezíti az azonosítást.

A collate függvény jelentősége az adatfeldolgozás során az adatok batch-ekbe szervezésében rejlik, amelyeket a modell egyszerre dolgoz fel. A collate függvény az adatok egyedi igényei szerint állítja össze a batch-eket, például a képek méretének, formátumának vagy az annotációk típusának figyelembevételével. Ez lehetővé teszi a dinamikus adaptációt, ha az adathalmaz változó dimenziókkal vagy metaadatokkal dolgozik, ami különösen fontos a FungiCLEF esetében, ahol a metaadatok kulcsszerepet játszanak az ökológiai kontextus modellezésében.

3. Módszertan

Manapság már egy általánosan bevett módszer, hogy képosztályozási feladatokra felhasználjunk egy már prior, képekre betanított modellt, és ezt finomhangoljuk. Gondoljunk bele, hogy egy alapoktól implementált keretrendszer csak akkor tud kiemelkedő teljesítményt nyújtani, ha megfelelő mennyiségű adathalmaz áll rendelkezésre, ami legtöbb esetben meghiúsul. A gyorsabb konvergenciát segíti, ha a klasszifikáló már "hozzá van szokva a képekhez", azaz kellő mennyiségű adatot látott az új feladat előtt. Ez a gondolat visszavezethető a világmodellekre, ahol képek sokasága feleltethető meg a világ működéséről alkotott belső modellnek [16].

3.1. Legkorszerűbb megközelítések

Jelenleg Christopher Chiu és társai (2024) [2] vezetik a klasszifikációs toplistát Ábra. 1. Két megközelítést vizsgálták: (1) egy számítógépes látásmodell teljes körű betanítása, és (2) egy osztályozó fej betanítása előre számított beágyazásokon. Míg az első megközelítés a számítógépes látásfeladatok hagyományos módszere, addig a második jelentősen kevesebb memóriaigénnyel és gyorsabb betanítási idővel jár. A legjobban teljesítő modelljük egy ensemble modell volt, amely a DINOv2 beágyazásokon alapult, és két osztályozó fejjel rendelkezett, amelyeket a kétfázisú cross-validation tanítás során hoztak létre. A metaadatok kiegészítő predikciós célként való felhasználását is vizsgálták, bár a metaadatok bevonása némi marginális javulást mutatott a validációs pontosság és az F1-érték tekintetében. Ez összhangban van a korábbi kutatások eredményeivel, amelyek szerint a metaadatok inputként való beépítése pozitívan befolyásolja a modell teljesítményét [2]. Ezek alapján választottunk megközelítéseket, megtartva a keretrendszer modularitását és skálázhatóságát. Négy különböző modellt implementáltunk és hasonlítottuk össze melyek főleg backbone felépítésükben tértek el, illetve abba, hogy használnak-e már előtanított súlyokat:

1. Conv model: A konvolúciós hálózatok erősek a lokális jellemzők és mintázatok felismerésében, különösen vizuális adatok esetén. Hatékonyan skálázhatók és egyszerű architektúrájukkal gyors feldolgozást biztosítanak nagy adathalmazokon is.

2. Attention model: A releváns jellemzők kiemelésével javítja a predikció pontosságát, különösen komplex mintázatok esetén.

3. Finetuned model: Előtanított modellek felhasználásával gyors konvergenciát és erős teljesítményt nyújt kisebb adathalmazokon.

4. Multi-task Metaformer alapú bináris osztályozó finomhangolása és tanítása a CAFormer [15] alapmodellt használatával, két külön fejre bontva, egyet a fajok osztályozására és egyet a toxicitás predikciójára, ezzel optimalizálva a többcélú tanulást. A modell ígéretes megközelítést mutat a metainformációk feature vektorokoban való eltárolásával, ami a jelen adathalmaznál nem kiemelkedő, de erőteljes skálázhatóságot biztosít, ha a fajok előfordulásainak és karakterisztikáinak sokkal nagyobb a szórása, akár földrajzi természetes élőhely szerint.

	Name	Track 1 ↓	Track 2 ↓	Track 3 ↓	F1 ↑	Acc. ↑
Private	MetaFormer (Competition)	0.391	1.604	2.044	30.0	60.9
Private	DINOv2 (Post Competition)	0.216	0.129	0.345	57.7	78.4
Public	MetaFormer (Competition)	0.395	1.649	2.044	27.6	60.5
Public	DINOv2 (Post Competition)	**0.211**	0.165	0.375	49.8	**79.0**
Public	Rank 1 - IES	0.2922	**0.0699**	**0.3621**	**54.99**	70.78
Public	Rank 2 - jack-etheredge	0.2394	0.1681	0.4075	49.81	76.06
Public	Rank 6 - Baseline with EfficientNet-B1	0.4926	0.6599	1.1526	32.99	50.74

1. ábra. Nyilvános és privát teszthalmaz pontszámok a hivatalos ranglistáról [2]

3.2. Adatfeldolgozás

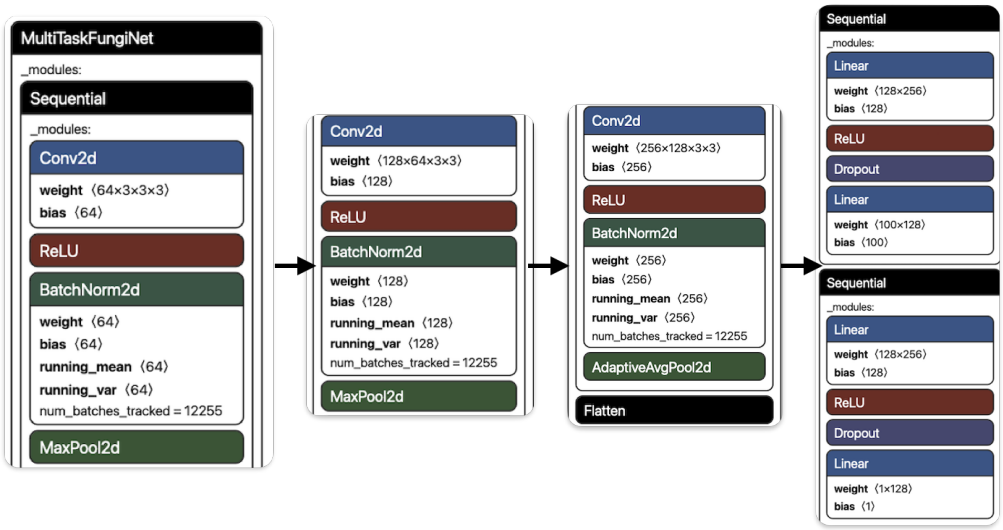
Az adatfeldolgozásunk konzisztens a keretrendszerek között és megszokott pytorch adatbetöltőkön alapul. Mivel a gombák vizuális hasonlósága és változatos morfológiája miatt a felismerés gyakran bonyolult, az adatokat előkészítése ezért fontos lépés, hogy a modellek megfelelően tudják kezelni őket. A képek felbontása és dimenziója különböző lehet, ezért fontos azokat egységes méretre átméretezni, hogy a modell egyenletesen tudja kezelni őket. Az interpolációs technikák alkalmazása ezen a ponton lehetővé teszi a képek sima átméretezését, miközben a lényeges információk nem veszítenek el. Az adatfeldolgozás másik fontos aspektusa a képek normalizálása. Mivel a képek különböző fényviszonyok között és különböző eszközökkel készülhettek, a normalizálás segít abban, hogy az adatok egységes skálára kerüljenek, így a modell ne legyen érzékeny a fényképezési körülmények különbségeire. A képek előkészítése után azok egyenként kerülnek betöltésre a modell számára, miközben a képek színcsatornáit is megfelelő formátumba alakítjuk, hogy a modell számára értelmezhetővé váljanak. A collate függvény biztosítja, hogy a bemenetek (képek, címkék, metaadatok stb.) megfelelően legyenek összegyűjtve és előkészítve a tanulási folyamat során. Az adatokat egy batch-be szervezi, figyelembe véve a különböző adatformátumokat, például a képek méreteit, típusait és az egyéb kiegészítő információkat, mint a címkék vagy a metaadatok.

3.3. AdaBoost és Gradient Boosting

A Gradient Boosting egy iteratív tanuláson alapuló ensemble technika, amely döntési fák kombinációjával minimalizálja az előrejelzési hibát és javítja a modell teljesítményét súlyozási mechanizmusokkal. Az AdaBoost hasonlóan gyenge osztályozók kombinációját alkalmazza, és a OneVsRest stratégiával hatékonyan kezeli a többosztályos problémákat, az egyes osztályokat külön bináris modellként kezelve. Mindkét architektúra jól skálázható, és pontos osztályozást biztosít a komplex adathalmazokon.
1. Gradient Boosting Classifier a mérgezőségi osztályozásra A Gradient Boosting Classifier (GBC) implementációja a ResNet-50 neurális hálózat által előállított jellemzővektorokat használja bemenetként. A ResNet-50 előre betanított mély neurális hálózat, amely kiváló vizuális reprezentációkat kínál, jelentősen csökkentve a nyers képadatok feldolgozási komplexitását. A GBC iteratív módon épít döntési fákat a ResNet-50 által generált magas dimenziójú és informatív jellemzőkön. Az osztályok közötti egyenlőtlen eloszlás kezelésére a pipeline kiszámítja az osztálysúlyokat. Ez biztosítja, hogy a mérgező osztályok alulreprezentáltságát megfelelő súlyozással kompenzáljuk a tanulási folyamat során. Az így számított súlyokat a **GradientBoostingClassifier** tanítási lépésében, mint sample weights alkalmazzuk, ami jelentősen javítja a modell predikciós teljesítményét a kisebbségi osztályoknál.
2. One-vs-Rest AdaBoost osztályozó a fajspecifikus osztályozásra A One-vs-Rest AdaBoost Classifier szintén a ResNet-50 által előállított jellemzővektorokat használja az egyes gombafajok azonosítására. Az AdaBoost algoritmus gyenge tanulók (döntési fák) iteratív súlyozásával biztosít magas osztályozási pontosságot. Az One-vs-Rest stratégia külön bináris osztályozót tanít minden faj számára, amely hatékonyan skálázódik nagyszámú osztály esetén. A ResNet-50 által nyert jellemzők lehetővé teszik a finom különbségek azonosítását a különböző gombafajok között, míg az AdaBoost algoritmus tovább optimalizálja az egyes fajokra vonatkozó teljesítményt.

3.4. Multi-head CNN model

Multi-head CNN model egy olyan konvolúciós neurális hálózat, amely a gombák képi adatai alapján végzi el a fajspecifikus osztályozást és toxicitási előrejelzést, lásd Ábra. 2. A modell egy könnyen skálázható CNN hátteret használ az alapvető jellemzők kinyerésére, majd a kinyert jellemzőket két különálló előrejelző fej dolgozza fel. Ez az architektúra hatékony és robusztus megoldást kínál többcélú tanulási feladatokhoz.



2. ábra. Multi-Head CNN arhitektúra

Komponens	**Leírás**
CNN Alapú Encoder	
Konvolúciós Réteg	Kernel méret: 3, Padding: 1. Szűri a bemeneti kép jellemzőit.
ReLU Aktiváció	Nemlinearitást ad a modellhez, lehetővé téve a komplex minták tanulását.
Batch Normalization	Stabilizálja a tanulási folyamatot, csökkentve a belső covariancia eltolódást.
MaxPooling Réteg	Csökkenti a térbeli méretet, miközben megőrzi a legfontosabb jellemzőket.
Adaptive Average Pooling Réteg	Fix méretű kimenetet generál az encoder végén.
Flatten Réteg	
Flatten Réteg	Az encoder kimeneti jellemzőit 1 Dimnziós vektorrá alakítja.

1. táblázat. CNN Alapú Encoder Architektúra és Flatten Réteg

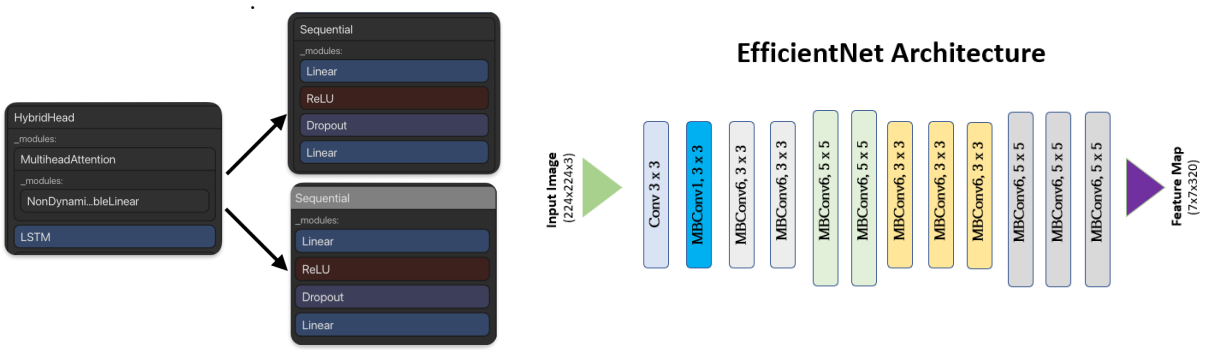
Komponens	**Leírás**
Multi-task Heads	
Fajspecifikus osztályozás	
Lineáris Réteg	A kinyert jellemzőket 128-dimenziós térbe transzformálja.
ReLU Aktiváció és Dropout (0.3)	Javítja a tanulás stabilitását és csökkenti az overfitting kockázatát.
Végső Lineáris Réteg	Az osztályok számának megfelelő dimenziójú kimenetet ad.
Toxicitási előrejelzés	
Hasonló felépítés	Hasonló az osztályozó fejhez, de a bináris probléma.
Forward Metódus	
Encoder feldolgozás	A bemeneti képeket az encoder feldolgozza, amely jellemzőket állít elő.
Flatten Réteg	Kisimítás.
Párhuzamos feldolgozás	A kinyert jellemzőket a két fej párhuzamosan dolgozza fel.
Kimenet	Osztályspecifikus és toxicitási előrejelzést.

2. táblázat. Multi-task Heads és Forward Metódus

Ez a modell (lásd. Táblázat 1, Táblázat 2) ideális a komplex képi adatok elemzésére, miközben figyelembe veszi mind a fajok szerinti osztályozás, mind a toxicitás becslésének sajátos követelményeit. A veszteségfüggvények kombinálása a fajspecifikus osztályozás és toxicitási előrejelzés céljaira CrossEntropyLoss és BCEWithLogitsLoss lehetővé tette a két feladat egyensúlyának kezelését. Az **Adam optimalizáló** használata gyors és hatékony konvergenciát biztosított, míg a tanulási ráta dinamikus csökkentése ReduceLROnPlateau a validációs veszteség alapján finomította a modell teljesítményét. A legjobb modell mentése a validációs veszteség alapján biztosította, hogy a legjobb általánosítási képességgel rendelkező modellt őrizzük meg.

3.5. EfficentNet-b0 finetuned Hybrid MHCNN

EfficientNet-b0 [6] alapú előképzett hátteret használja vizuális jellemzők kinyerésére, továbbá egy HybridHead (Ábra. 3) komponens segítségével ötvözi az Attention mechanizmus és az állapottér-modell (SSM) előnyeit. Az Attention réteg a bemeneti jellemzők közötti kapcsolatok kiemelésére szolgál, míg az SSM (LSTM-alapú) szekvenciális összefüggések feldolgozására képes, ezzel javítva a reprezentáció robusztusságát (Táblázat 3). A két különálló teljesen összekötött fej külön kezeli a fajspecifikus osztályozási és toxicitási előrejelzési feladatokat, biztosítva az optimalizált többcélú tanulást. Az EfficientNet hátterű modellek ismerten kiváló teljesítményt nyújtanak a képfeldolgozásban, míg az Attention-alapú architektúrák és az SSM-ek szinergikus használata új távlatokat nyit a komplex adatok többdimenziós kiértékelésében (Táblázat 4) [12].



3. ábra. Bal oldalon a Hibrid Multihead achitektúra, jobb oldalon az EfficientNet architektúrája látható

Komponens	**Leírás**
Attention Mechanizmus	MultiheadAttention réteg, amely a jellemzők közötti összefüggésekre fókuszál.
Állapottér-modell (SSM)	LSTM réteg, amely a minták feldolgozását végzi a figyelmi kimenetek alapján.
Fajspecifikus osztályozás	Két teljesen összekötött réteg, 512 rejtett neuronnal, ReLU aktivációval és 0.5 Dropout-tal.
Toxicitás előrejelzés	Hasonló felépítés, de a kimenet dimenziója a toxicitási osztályozásra igazított.
Forward metódus lépései	Figyelmi mechanizmus az összefüggések kiemelésére. Állapot tér feldolgozás.Globális átlag pooling az információ aggregálására majd osztályozás és toxicitási előrejelzés.

3. táblázat. HybridHead Architektúra Összefoglalása

A tréning során a veszteségfüggvény súlyozását (alpha és beta) alkalmaztuk a fajspecifikus és toxicitási osztályozás egyensúlyának biztosítására. **Early stopping** került bevezetésre, hogy megakadályozza a túlillesztést, míg a legjobb validációs pontosságú modell mentésével biztosítottuk a legjobb paraméterek megőrzését. Ezek a technikák célzottan javítják a modell stabilitását és általánosítási képességét.

Komponens	Leírás
EfficientNet-B0 Backbone	Az EfficientNet-B0 egy könnyű, de nagy teljesítményű neurális hálózat, amely előképzett súlyokat használ az ImageNet adathalmazról. Fő célja a bemeneti képekből a legfontosabb vizuális jellemzők hatékony kinyerése, alacsony számítási igény mellett.
Hibrid-Multihead component	A MultiheadAttention réteg kiemeli a vizuális jellemzők közötti releváns kapcsolatokat, így segítve a fontos minták azonosítását.
Forward metódus lépései	A backbone kinyeri a vizuális jellemzőket. A HybridHead feldolgozza ezeket: figyelmi mechanizmus, állapottér-feldolgozás és globális pooling után két kimenetet generál: osztályozás és bináris becslés.

4. táblázat. EfficientNet-b0 Finetuned Hybrid MHCNN Modell Komponensei

3.6. Multitask-CAFormer

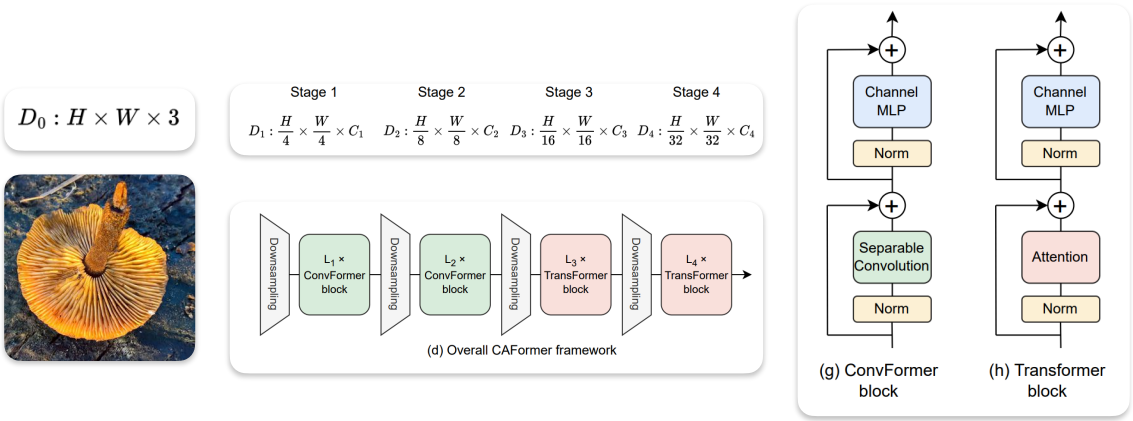
A FungiCLEF kihívás során egy multitask architektúrát implementáltunk a CAFormer modell alapján, hogy hatékonyan végezzük el a gombafajok osztályozását és a mérgezési tulajdonságok bináris predikcióját. A CAFormer modell kifejezetten alkalmas erre a feladatra, mivel képes különböző típusú információk egyidejű feldolgozására a hierarchikus architektúrájának és fejlett token mixer mechanizmusainak köszönhetően.
A Timm (PyTorch Image Models) könyvtár egyszerű és gyors hozzáférést biztosít a legmodernebb mélytanulási modellekhez, beleértve a CAFormer architektúrát is, amely a MetaFormer keretrendszerre épül. Az alábbi kódrészlet bemutatja, hogyan inicializálható a CAFormer modell a Timm segítségével:

```
base_model = timm.create_model(
    "caformer_s18.sail_in22k",
    pretrained=True,
    num_classes=num_species_classes
)
base_model.to(device)
```

A modell alapja a MetaFormer architektúra, amelyet mélytanulási kontextusban univerzálisan alkalmazható keretrendszerként definiálnak Ábra. 4, Táblázat 5. A MetaFormer blokkok és a CAFormer a token mixing mechanizmusok optimalizálásával ér el kiemelkedő teljesítményt, amelyek a helyi és globális mintázatok hatékony felismerését biztosítják. Az alsó rétegekben szeparálható konvolúciókat (SepConv), míg a felső rétegekben vanilla self-attention mechanizmust alkalmaz. [15].

Rész	Komponensek	Funkció
Stem	Conv2d, LayerNorm2dNoBias	Bemeneti képek feldolgozása, jellemzők kinyerése
Downsampling	Konvolúciók, Átméretezés	Dimenziók csökkentése, információ tömörítése
ConvFormer block	SepConv, Channel MLP	Lokális mintázatok felismerése és feldolgozása
Transformer block	Attention, Channel MLP	Globális összefüggések és mintázatok tanulása
MetaFormer Stage	Több blokk (Conv/Transformer)	Hierarchikus jellemzők létrehozása
Head	AvgPool, Linear	Osztályozási döntések a tanult jellemzők alapján

5. táblázat. A CAFormer architektúra fő komponensei és funkciói



4. ábra. A CAFormer általános keretrendszere. (d) Négy szakaszból áll, ahol az első két szakasz ConvFormer blokkokat tartalmaz a lokális mintázatok feldolgozására, míg az utolsó két szakasz Transformer blokkokkal globális összefüggéseket tanul. A dimenziócsökkentést minden szakasz előtt konvolúciós rétegek végzik. (g) A ConvFormer blokk szeparálható konvolúciókat és csatorna-alapú MLP-ket használ, hogy hatékonyan dolgozza fel a lokális mintázatokat. (h) A Transformer blokk self-attention mechanizmus segítségével globális összefüggéseket tanul, és MLP rétegekkel egészíti ki a feldolgozást [15].

Az alapmodellre építettünk rá egy multitask fejet, ami két feladatot lát el:

1. **Gombafajok osztályozása:** A CAFormer alapmodell egy osztályozó fejjel egészül ki, amely az előszámított jellemzők alapján végzi el a 100 faj kategorizálását. Az architektúra globális átlag pooling réteget alkalmaz, amely biztosítja az információk tömörítését és a modellezés egyszerűsítését.
2. **Toxicitás predikciója:** Egy második osztályozó fej bináris osztályozást végez a mérgező és ehető tulajdonságok azonosítására. Ehhez a globálisan poolingolt jellemzőkből egy egyszerű teljes összeköttetésű (fully connected) réteg biztosítja a predikciót.

A modell implementációja során előre betanított paraméterekkel inicializáltuk a CAFormer-t, ezzel minimalizálva az edzési időt és a memóriaigényt. Az optimalizáláshoz Adam algoritmust alkalmaztunk, 0,0001-es tanulási rátával. Az edzési folyamatot két különböző veszteségfüggvénnyel irányítottuk: a CrossEntropyLoss-t alkalmaztuk mind a fajok osztályozására, mind pedig a toxicitás predikciójára. Az egyes veszteségkomponenseket súlyozva kombináltuk ($\alpha = 1.0 - \beta = 0.6$), hogy a fajok osztályozása prioritást élvezzen, amiből következtethető lehet a mérgezettség is.

3.7. Implementáció

A megvalósításokat pytorch környezetben végeztük és jupyter lokális kernel illetve colab környezetben futtattuk. Mindegyik modell tanítható Machintosh M2 és M1 cpu-n, de hatékonyabb és gyorsabb konvergencia miatt a Metformerhez egy A100 gpu-t használtunk. A teljes kódbázis megtalálható és nyitott forráskóddal elérhető a következő repozitóriumban: https://github.com/CIMBIBOY/FungiCLEF2024_ADC. A négy modellünk tanulítása rendere: adagrad-, MHCNN-, caformer-, caformer_continue-, eval-.ipynb fájlokban található a main branchen. A tanítások futtatásához csak egy Jupyter nootbook környezet szükséges.

4. Results

4.1. Evaluation criteria

Az eredeti versenyben négy egyedi mérőszám mellett szerepelt a makro-átlagolt F1 pontszám és a pontosság. Mivel a mi feladatunk egyszerűbb volt, és nem igényelte az ismeretlen fajok előrejelzését, az értékelési metrikákat egy egyedi mérőszámra, a makro-átlagolt F1 pontszámra, recall és az osztályozási hibára szűkítettük, amelyet a pontosságból következtettünk.

4.2. GradientBoost and AdaBoosting

A **GradientBoostClassifier** kiemelkedően teljesít a nem mérgező osztályban, de a mérgező osztály magas false negative rátája jelentős kockázatot jelent, míg az **AdaBoostClassifier** alacsony általános pontossága ellenére néhány kiegyensúlyozott supporttal rendelkező osztályban mutatott jó teljesítményt GradientBoostClassifier mérgezettségre (Táblázat 6, Táblázat 7):

Class	**Precision**	**Recall**	**F1-Score**	**Support**
Non-Poisonous (0)	0.95	0.89	0.92	4722
Poisonous (1)	0.31	0.54	0.39	430
Accuracy	0.86			
Macro Avg	0.63	0.71	0.66	5152
Weighted Avg	0.90	0.86	0.88	5152

6. táblázat. GradientBoostClassifier metrikák mérgezettségre

	Predicted: Non-Poisonous (0)	**Predicted: Poisonous (1)**
Actual: Non-Poisonous (0)	4197	525
Actual: Poisonous (1)	198	232

7. táblázat. GradientBoostClassifier kovarianciamátrix mérgezettségre

OneVsRest- AdaBoostClassifyer használata a fajspecifikus osztályozásra (Táblázat 8):

Metric/Class	**Precision**	**Recall**	**F1-Score**	**Support**
Overall Accuracy	0.34			
Macro Average	0.35	0.34	0.34	5152
Weighted Average	0.36	0.34	0.34	5152
Class 5 (High F1)	0.79	0.57	0.66	46
Class 19 (High F1)	0.66	0.50	0.57	70
Class 31 (Balanced)	0.69	0.57	0.63	63
Class 79 (Notable)	0.72	0.63	0.67	49
Class 99 (Balanced)	0.52	0.54	0.53	56

8. táblázat. AdaBoostClassifyer metrikák gombafaj osztályozásra

4.3. Multi-headed CNN

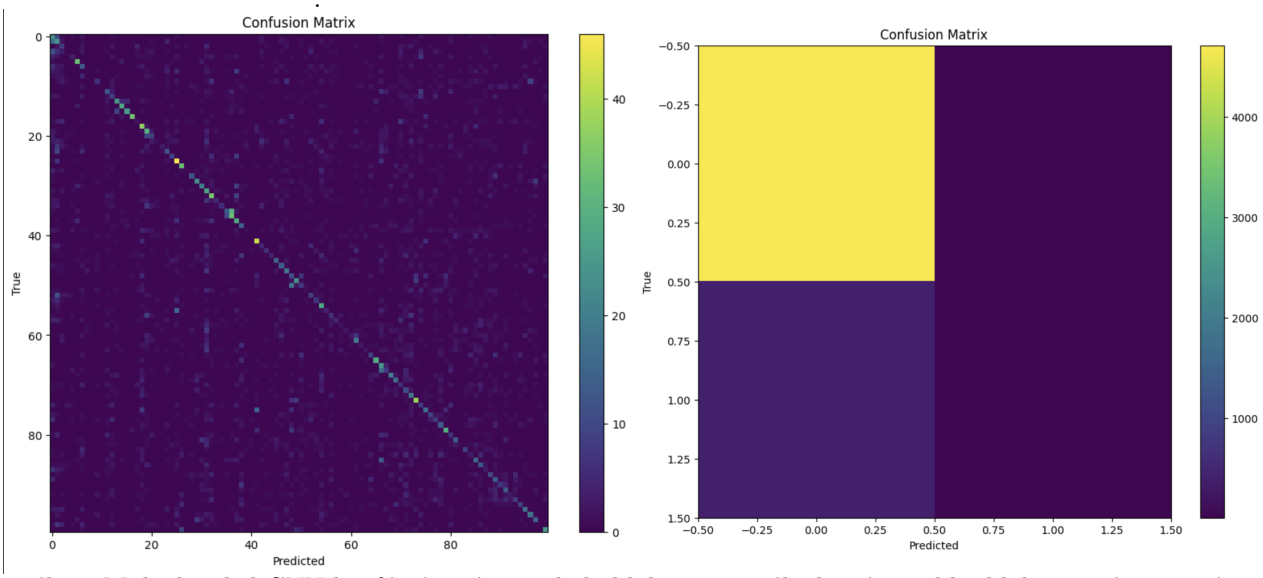
A **Multi-headed CNN** (Táblázat 9, Táblázat 14, Ábra. 5) modell a fajspecifikus osztályozásnál alacsony, 26%-os pontosságot ért el, amit az osztályok közötti egyenlőtlen eloszlás magyaráz. A toxicitás osztályozásban jobb teljesítményt mutatott, 92%-os pontossággal. A modell javításához kiemelten fontos a mérgező osztály súlyozása vagy az adatok kiegyensúlyozása.

Metric/Class	**Precision**	**Recall**	**F1-Score**	**Support**
Overall Accuracy	0.26			
Macro Average	0.28	0.26	0.24	5153
Weighted Average	0.28	0.26	0.24	5153
Class 5 (Highest F1)	0.61	0.72	0.66	46
Class 18 (Balanced)	0.29	0.63	0.39	59
Class 25 (Significant)	0.36	0.58	0.45	79
Class 99 (Weighted)	0.45	0.45	0.45	56

9. táblázat. MHCNN eredményei a gombafajokra

Class	**Precision**	**Recall**	**F1-Score**	**Support**
Non-Toxic (0)	0.92	1.00	0.96	4722
Toxic (1)	0.73	0.06	0.10	430
Accuracy	0.92			
Macro Avg	0.82	0.53	0.53	5152
Weighted Avg	0.90	0.92	0.89	5152

10. táblázat. MHCNN eredményei a mérgezettségre



5. ábra. Multi-headed CNN konfúziós mátrix a bal oldalon at osztályokra és a jobb oldalon a mérgezettségre

4.4. EfficentNet-b0 finetuned Hybrid MHCNN eredmények

Eredményeken látható, hogy a fajspecifikus osztályozásban az átlagos pontosság 35%, míg a toxicitási predikcióban kiemelkedő, 94%-os pontosságot ért el (Táblázat 11). A toxicitás osztályozásában a non-poisonous osztály precision és recall értékei egyaránt 97%, azonban a poisonous osztály esetében csak 0.62%-ban találta el, hogy ténylegesen mérgező volt (Táblázat 12). Ez különösen kritikus, mivel el kell kerülni, hogy mérgező gombát fogyaszthatónak osztályozzunk (Ábra. 6).

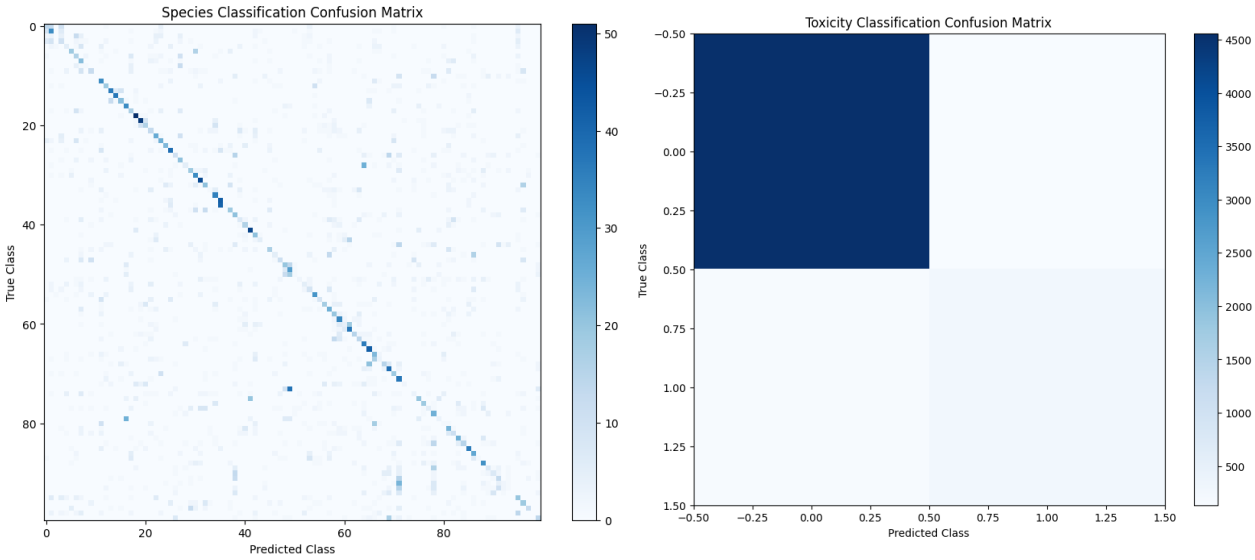
11. táblázat. Hibrid-Multihead CNN eredményei osztályozásra

Metrika/Osztály	**Precision**	**Recall**	**F1-Score**	**Support**
Legjobb teljesítmény	0.79	0.90	0.84	Class 14 (41)
Legalacsonyabb teljesítmény	0.00	0.00	0.00	Több osztály (pl. 2, 10, 73)
Átlag (macro)	0.34	0.35	0.31	5120
Átlag (weighted)	0.34	0.35	0.32	5120
Teljes pontosság (accuracy)	0.35			

12. táblázat. Hibrid-Multihead CNN eredményei mérgezettségre

Osztály	**Precision**	**Recall**	**F1-Score**	**Support**
Non-Poisonous (0)	0.97	0.97	0.97	4693
Poisonous (1)	0.67	0.63	0.65	427
Accuracy	0.94			
Macro Avg	0.82	0.80	0.81	5120
Weighted Avg	0.94	0.94	0.94	5120

.



6. ábra. Konfúziós mátrix az osztályokra és jobb oldalon a mérgezettségre Hybrid-Multihead CNN modellel

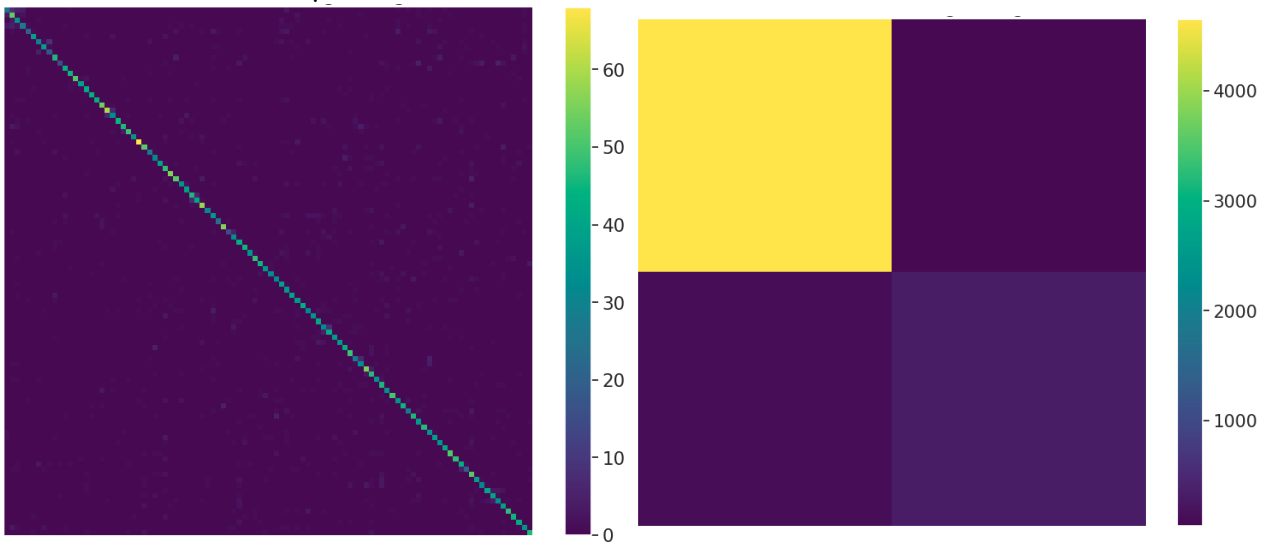
4.5. Multitask Metaformer Results

A CAFormer alapú modell egy dolgozóval, tanítás közben töltötte ba ez adatokat és közel 42 + 4 + 4 + 1 iterációig tanult, A100 high-ram gpu-n 256-os batchel, majd egy L4 gpu-n 128-as batchel és végül egy T4 gpu-n 64-es batchel. A Metaformer irányú megközelítés bizonyult a legjobbnak (lásd. Táblázat 13, Táblázat 14, Ábra. 8), hiszen közel 16 iteráció után elérte a 74%-os fajalapú és 90%-os mérgezettség alapú pontosságot. Ez később csak egy keveset változott, ezzel kiemelkedve a modellek között (lásd. Ábra. 7). Az eredmények hiába jobbak, mint a hivatalos ranglista természetesen nem összehasonlítható, hiszen csak 100 fajra történt az osztályozás. Minden esetre ennyi, relatívan véletlenszerűen kiválasztott faj esetén ez egy elég jó eredmény. Mivel a metainformációk ezen feladat esetén nem vezettek volna lényeges javulásra, így ezek használatára nem tértünk ki, de az implementáció könnyen megvalósítható.

Epoch 51/51 Summary:

Metric	Value
Train Loss	0.0109
Train Species Accuracy	99.67%
Train Toxicity Accuracy	100.00%
Validation Loss	1.3235
Validation Species Accuracy	78.28%
Validation Toxicity Accuracy	97.01%

7. ábra. A Multitask CAFormer végső pontossága a klasszifikációs feladatra.



8. ábra. Az alábbi képeken (hasonló skála értékekkel, mint a korábbi ábráknál) a Multitask CAFormer konfúziós mátrixa látható bal oldalon osztályokra és jobb oldalon a mérgezettségre.

Class	Precision	Recall	F1-Score	Support
Best Class (79)	0.89	0.96	0.92	49
Worst Class (0)	0.73	0.47	0.57	51
Accuracy		0.78		
Macro Avg	0.79	0.78	0.78	5120
Weighted Avg	0.79	0.78	0.78	5120

13. táblázat. Klasszifikációs táblázat gombafajokra MT-Metaformer alapú megközelítéssel.

Class	Precision	Recall	F1-Score	Support
Non-Toxic (0)	0.98	0.99	0.98	4693
Toxic (1)	0.88	0.74	0.80	427
Accuracy	0.97			
Macro Avg	0.93	0.86	0.89	5120
Weighted Avg	0.97	0.97	0.97	5120

14. táblázat. Klasszifikációs táblázat mérgezettségre MT-Metaformer alapú megközelítéssel.

4.6. Ablációs vizsgálat gombaklasszifikációra

Model	Class	Precision	Recall	F1-Score	Support
GradientBoostClassifier & Adaboosting	Non-toxic (0)	0.95	0.89	0.92	4722
	Toxic(1)	0.31	0.54	0.39	430
	Toxicity Accuracy	0.86			
	Species Accuracy	0.34			
Multi-headed CNN	Non-toxic (0)	0.92	1.00	0.96	4722
	Toxic(1)	0.73	0.06	0.10	430
	Toxicity Accuracy	0.92			
	Species Accuracy	0.26			
Hybrid-Multihead CNN	Non-toxic (0)	0.97	0.97	0.97	4693
	Toxic(1)	0.67	0.63	0.65	427
	Toxicity Accuracy	0.94			
	Species Accuracy	0.35			
Multitask Metaformer	Non-toxic (0)	0.98	0.99	0.98	4693
	Toxic(1)	**0.88**	0.74	**0.80**	427
	Toxicity Accuracy	**0.97**			
	Species Accuracy	**0.78**			

15. táblázat. **Gombafajok osztályozásának összehasonlítása a négy modellünk alapján.**

A különböző modellek eltérő erősségekkel és gyengeségekkel rendelkeztek a gombafajok osztályozásában, amit az ablációs Táblázat 15 is mutat. A **GradientBoostClassifier & AdaBoosting** modellek egyszerűségük és hatékonyságuk miatt kiemelkedően teljesítettek a nem mérgező osztályban, de a mérgező osztály gyenge recall értéke korlátozta használhatóságukat. A **Multi-headed CNN** magas mérgezettségi pontosságot mutatott a nem mérgező osztályban, ugyanakkor a mérgező osztály recall értéke nagyon alacsony maradt, ami jelentős hátrány. A **Hybrid-Multihead CNN** modell javított a mérgező osztály recall értékén, de teljesítménye továbbra sem érte el a legjobb szintet.

A **Multitask Metaformer** modell mind a mérgezettség, mind a fajspecifikus osztályozásban kimagasló pontosságot ért el, kiegyensúlyozott teljesítményt nyújtva a két osztály között. Ez a transformer-alapú architektúra a legjobb eredményeket hozta, és különösen alkalmas lehet további fejlesztésre nagyobb és kiegyensúlyozottabb adathalmazokon, bár számítási igénye magasabb a többi modellhez képest.

5. Konklúzió és Továbbfejlesztések

A Metafromer [14] különös előnye a metaadatokban rejlik, ugyan a jelenlegi adathalmazban a gombafajok honosságának geokoordinátája és jellemző éghajlata Dániára szűkített, így igazi előnye csak nagyobb adathalmazok esetén mutatkozna. Ez persze kiterjeszthető gombákkal az egész világról, esetlegesen meta-információkkal vagy korrelációs adatok hozzáadásával. Illetve bővíthető gombáktól eltérő fajokra, hiszen a metaformer modellek, biztosítják az ehhez szükséges modularitást.

Munkánkban bemutattunk egy részletes ablációs vizsgálatot, melyben összehasonlítottunk négy potenciális modellt a gombaosztályozási feladatra. Mind a négy architektúra javítható lenne az adathalmaz növelésével és metainformációk felhasználásával. Az célul kitűzött modularitás és skálázhatóság sikerült, hiszen mindegyik modell kiegészíthető, komponenseik lecserélhetőek és a használt alapmodell variálható. Legjobb eredményünk a Multitask CAFormerrel született, mely 78%-os pontosságot ért el fajosztályozásban és 97%-os pontossággal becsülte a mérgező fajokat. Az ablációs kutatás egyértelműen rámutat arra, hogy még a legkorszerűbb architektúrák is jelentős kihívásokkal szembesülnek a feladat megoldása során.

5.1. Csapatmunka

A feladatmegoldást egyenlő részekre osztottuk, Ágnes foglalkozott az adathalmazzal, Márk fogalmazta az általános tanítási folyamatot és tanította a Metaformer architektúrát és Dávid implementálta a másik három modellt amit a kutatás során megvizsgáltunk. Az eredményekben egyenlő részesedéssel osztozunk.

Hivatkozások

[1] Bahram et al. Fungi as mediators linking organisms and ecosystems. *FEMS Microbiology Reviews*, 46(2):fuab058, 2022.

[2] Chiu et al. Fine-grained classification for poisonous fungi identification with transfer learning, 2024.

[3] Dosovitskiy et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[4] Frøslev et al. Danish mycological society, fungal records database, 2019.

[5] Lars Schmarje et al. Is one annotation enough? a data-centric image classification benchmark for noisy and ambiguous label estimation, 2022.

[6] Mingxing Tan et al. Efficientnet: Rethinking model scaling for convolutional neural networks, 2020.

[7] Picek et al. Overview of fungiclef 2023: Fungi recognition beyond 1/0 cost. In *Conference and Labs of the Evaluation Forum*, 2023.

[8] Qi Jia et al. Visual perception in text strings, 2024.

[9] Tan et al. Efficientnetv2: Smaller models and faster training. *arXiv preprint arXiv:2104.00298*, 2021.

[10] Tongyue Shi et al. Mathematical modeling analysis and optimization of fungal diversity growth, 2022.

[11] Xiaodan Liang et al. Computational baby learning, 2015.

[12] Xin Dong et al. Hymba: Hybrid-head architecture boosts small language model performance. *NVIDIA Technical Blog*, November 2024. Accessed: 2024-12-03.

[13] Yang et al. Treatment learning causal transformer for noisy image classification. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, page 6128–6139. IEEE, January 2023.

[14] Yu et al. Metaformer is actually what you need for vision. *arXiv preprint arXiv:2111.11418*, 2021.

[15] Yu et al. Metaformer baselines for vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(2):896–912, February 2024.

[16] Yann LeCun. A path towards autonomous machine intelligence, June 2022. Version 0.9.2.