

Celluláris hullámszámítógépek

Anime és nyugati rajzfilm megkülönböztetése képről (saját)

Projektfeladat megoldása

Ekart Csaba [ZWPMKP]

2020. április 30.

1. Bevezetés és futtatás

Ebben a jegyzőkönyvben a Celluláris Hullámszámítógépek tárgyra készített projektmunkámat mutatom be. Az elkészített programkód az alábbiak szerint futtatható:

Előkészületek:

- A mellékelt `anime_cartoon_cnn.ipynb` fájlt beimportáljuk Google Colabba, vagy az `anime_cartoon_cnn.py`-ban található kódot bemásoljuk.
- A mellékelt `dataset.zip` fájlt fel kell tölteni Google Colabba, majd kicsomagolni a `/content` helyre.
- A mellékelt `test.zip` fájlt azokat a képeket tartalmazza, amelyeken a végén az ellenőrzést végeztem. Az ezt tartalmazó cella futtatásához ennek feltöltése és kicsomagolása is szükséges.

Megjegyzés Lehet, hogy a jegyzőkönyvben bemutatott eredmények némileg eltérnek a Python Notebook fileban szereplőktől a többszöri futtatás és tesztelés miatt.

2. Probléma ismertetése

A személyes projektmunkám keretében egy olyan feladatot választottam, melyben modern neurális hálózatokon alapuló rendszerekkel próbálok osztályozni animékből, illetve hagyományos nyugati rajzfilmekből vett képeket. A feladat sokszor a laikus szemnek sem teljesen triviális, az sem teljesen egyértelmű, hogy pontosan mi különbözteti meg a két a kategóriát. De mi is pontosan az anime és miben különbözik az általunk megszokott rajzfilmes világtól?

2.1. Röviden az animékről (1) (2)

Az anime szó az angol *animation* szóból ered, és ez a hivatalos elnevezése a rajzfilmeknek Japánban, azonban világ minden más táján a Japánban készült rajzfilmeket értjük alatta. Műfaját tekintve rendkívül szerteágazó lehet, akciótól kezdve romantikus kalandokon keresztül a drámákig bármit találhatunk. De mi jellemző az ábrázolás módjára, amitől az átlag ember számára is könnyen megkülönböztethető a hagyományos rajzfilmek-től? Az animék látványvilága egyszerre táplálkozik a hagyományos japán fametszetekre és papírtekercekre rajzolt történetekből, illetve az akkoriban virágzó amerikai képregényműfaj ábrázolásmódjából. Ennek megfelelően az animékben a karakterek ábrázolása a nyugati rajzfilmekhez képest realiztikusabb, jellemző a karakterekre a nagy szem, kis száj, karcsúság, illetve az elnagyolt érzelem ábrázolás.



(a) Kép a *Dragon Ball Z* című animéből



(b) Kép a *Johnny Bravo* című rajzfilmből

1. ábra. Anime és hagyományos rajzfilm összevetése

3. A hálózat megvalósítása és felépítése (3) (4) (5) (7) (10)

A probléma megoldásához konvolúciós neurális hálózatot (CNN-t) használtam. A konvolúciós hálózatok előnye a hagyományos neurális hálózatokhoz képest, hogy míg utóbbi csak rendkívül egyszerű képek felismeréséhez használható (annak egyszerű 1D-s oszloppá alakításával), addig előbbi jóval nagyobb komplexitású képek osztályozására is alkalmas, mivel végig "eredeti" mátrixos formában használhatja az inputot. CNN használatával továbbá a képen található lokális tulajdonságok a megfelelő szűrők segítségével kiemelhetőek, így könnyítve a hálózat munkáját.

A megoldásomat pythonban készítettem a Tensorflow könyvtár és a Keras API segítségével Google Colabban.

3.1. Tanuló halmaz

Hogy a tanulás eredményre vezessen szükségem volt egy jó minőségű, kellően nagy méretű, felcímkézett adathalmazra. Mivel ehhez nem találtam olyat az interneten, amely alkalmas lett volna a hálózat tanításához, ezért saját képeket gyűjtöttem. Ez bizonyult a legfárasztóbb és leghosszadalmasabb feladatnak mindkét oldalról.

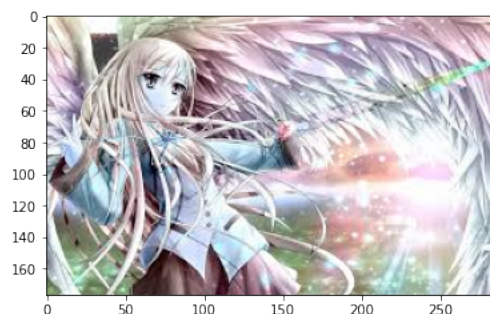
Ehhez a Google képkeresőt, illetve a *Batch Image Downloader* (6) nevű Firefox kiegészítőt használtam, hogy egyszerre nagy mennyiségű képet egyszerűen tudjak elmenteni. A képek kiválasztásánál szempont volt, hogy a műfaj egyértelműen eldönthető legyen laikus ember számára is.

A képeket cím szerint kerestem, igyekeztem olyan címek közül válogatni, ami mindkét kategória legikonikusabb elemeit tartalmazza, egyértelműen eldönthető róluk hogy melyik csoportba tartoznak, különböző látványvilággal rendelkeznek. A teljes tanuló halmaz mellékelve van a jegyzőkönyvhöz (5734 db kép), melyben kategória és cím szerint vannak rendezve a fájlok. Rajzfilm oldalon többek között olyan művek szerepelnek, mint a Looney Tunes, a Spongeyabob vagy a Tom és Jerry, míg animék között a Naruto, a Death Note, illetve a Dragon Ball.

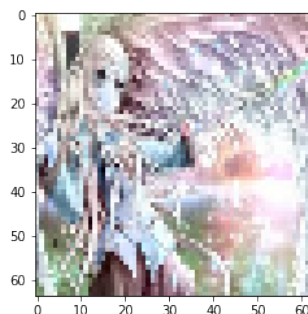
3.2. Képek előkészítése

Először az összes képet azonos méretre kell transzformálni, amely lehetőleg minél kisebb (a határán van annak, hogy egyértelműen felismerhető legyen a képről annak az osztálya). Erre azért van szükség, mert a hálózat csak fix bemenetű képeket tud fogadni, illetve minél kisebb képeket használunk annál többet javíthatunk a hálózat sebességén. Ehhez a `cv2.resize()` függvényt lehetett használni. Mivel ezeknél a képeknél meghatározó lehet a színezés, ezért mind a három színsatornát használni fogjuk.

Néhány példát végigpróbálva arra jutottam, hogy a 64x64-es képméret kielégítő lesz.



(a) Eredeti kép a tanulóhalmazból



(b) Átméretezett kép

2. ábra. Képek átalakítása

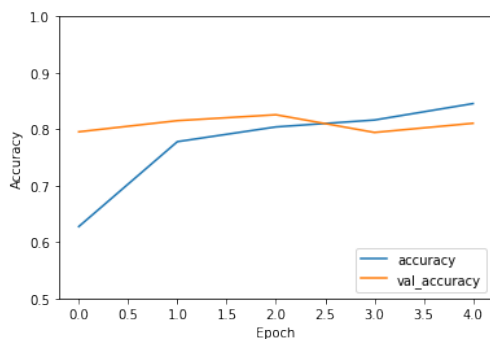
A mentett fájlokat mappában rendezve feltöltöttem a Google Colabba, majd egy egyszerű függvény segítségével elvégeztem rajtuk az átalakításokat, illetve kategóriájukkal együtt egy listába töltöttem őket. A listába töltés végén fontos, hogy az elemeket megkeverjük, hogy ezzel is csökkentsük az overfitting esélyét, és növeljük az adatok eloszlását. (7) (8)

3.3. Architektúra

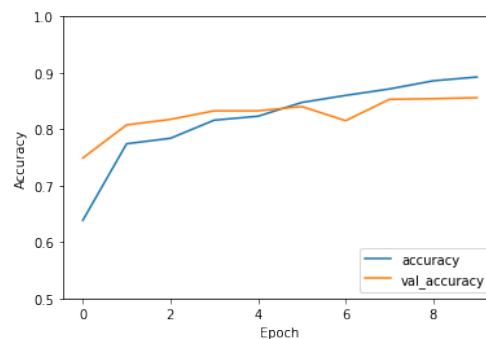
A hálózathoz egy egyszerű konvolúciós architektúrát alkalmaztam (5), amely 3 konvolúciós rétegből, közöttük max_pooling-gal, a végén két sima egyszerű Dense réteggel. Az összes layerben ReLU-t használtam aktivációs függvényként, kivéve az utolsónál, ahol szigmoidot, a bináris klasszifikáció miatt. A modellben 0.5-ös arányú dropoutot is alkalmaztam, hogy elkerüljem az overfittinget. Loss függvénynek a binary_crossentropy-t optimizernek meg szokás szerint ADAM-et választottam. Az architektúráról készült diagramot a Keras plot_model könyvtárának segítségével kiexportáltam. (9) (Ábra 3) A tanítás során az adatok 30%-át validálásra használtam fel, így ellenőrizve a hálózat tényleges tudását.

4. Eredmények

A hálózat eredményesen futott a megadott tanulóhalmazon és látszólag sikerült használható tudást elsajátítania a folyamat során. A tanulás végére a teszhalmazon kb. 81.1%-os accuracyt értem el. Összesen öt epochot futtattam csak, megfigyeléseim alapján ezután már overfittelni fog a hálózat, legalábbis jelentősen nem javul a teljesítménye.

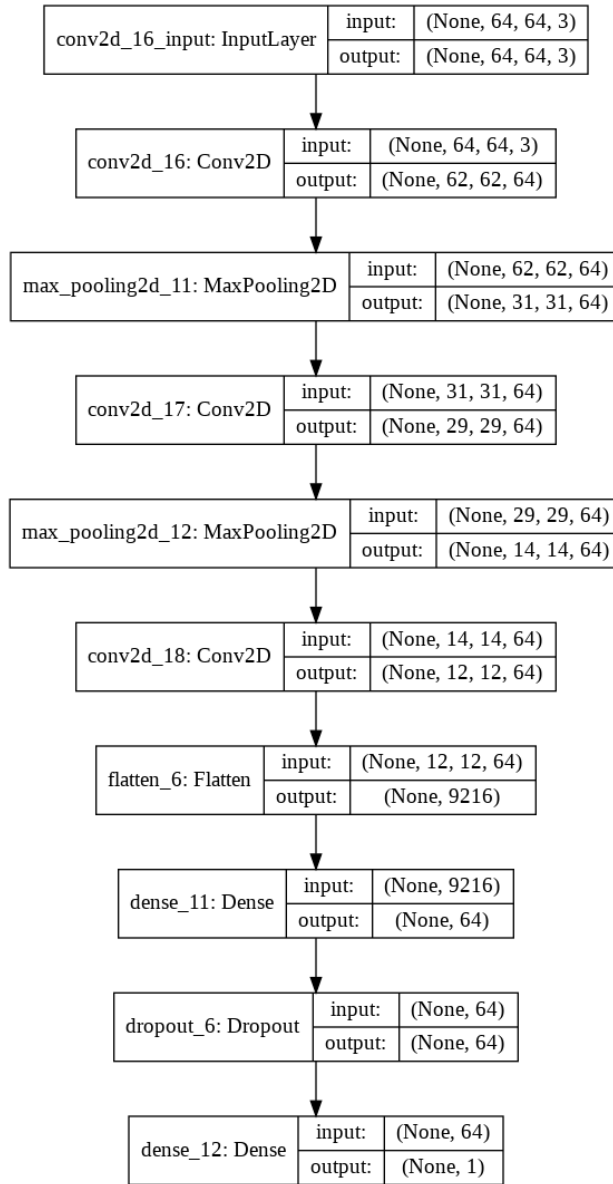


(a) Pontosság 5 epoch esetén



(b) Pontosság 10 epoch esetén

4. ábra. A hálózat pontosságának alakulása a tanuló (kék) és a teszt (narancs) halmazon



3. ábra. A hálózat architektúrája

4.1. Egyéb tesztek

A validáló halmazon kívül egy teljesen független teszt halmazon is megvizsgáltam a hálózat teljesítményét. Fő szempontom volt, hogy vajon képes-e felismerni olyan animét, amit még sosem látott, olyat ami meglepően hasonlít az általunk szokványosnak tekintett rajzfilmes műfajra, néhány magyar rajzfilm klasszikust, illetve olyan műfaji egyvelegeket, mint az *Aang legendája* vagy a később annak folytatásaként megjelent *Korra legendája*, melyek habár amerikai rajzfilmek, mind vizuálisan, mind történeti szempontból komolyan merítkezett az animék stílusából, és a keleti kultúrából.

Az alábbiakban ezen tesztek eredményeit veszem sorra:

4.1.1. "Könnyen eldönthető" példák

Először olyan képeken voltam kíváncsi az eredményekre, amelyeket egy átlag ember nagy bizonyossággal meg tud állapítani. Azt láthatjuk, hogy minden egyes példában sikeresen állapítja meg a műfajt, bár a "magabiztossága" nem tökéletes.



(a) Cartoon: 33.86% , Anime: 66.14%



(b) Cartoon: 14.95% , Anime: 85.05%



(a) Cartoon: 94.62% , Anime: 5.38%



(b) Cartoon: 89.16% , Anime: 10.84%

4.1.2. Érdekes példák

De mi a helyzet a bonyolultabb problémákkal? Amik a laikus szemnek nem egyértelműek már, hogy pontosan hova tartoznak. Ilyen példának hoztam a korábban említett *Aang legendája* és *Korra legendája* sorozatot. Az előbbi több képen is egyértelműen rajzfilmmnek vette, utóbbinál viszont a vártak szerint, elég bizonytalan döntést hozott.

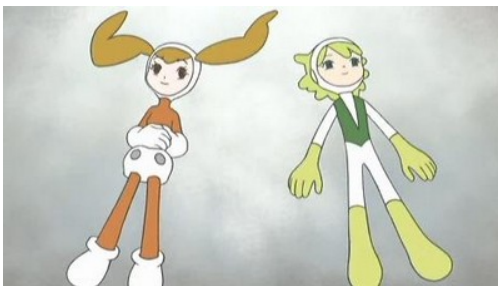


(a) Cartoon: 88.64% , Anime: 11.36%



(b) Cartoon: 43.46% , Anime: 56.54%

Egy másik érdekes példa a *Kaiba* című anime, aminek meglehetősen gyermeketeg, minimalista ábrázolás módja van. Ahogy azt vártam, a modell helytelenül a rajzfilm kategóriába sorolta többször is.



(a) Cartoon: 86.09% , Anime: 13.91%



(b) Cartoon: 66.2% , Anime: 33.8%

5. Végszó

Látható, hogy manapság egyszerű konvolúciós hálózatokkal rendkívül gyorsan létrehozhatunk olyan hálózatokat, melyek ilyen, és ehhez hasonló klasszifikációs problémák megoldására képesek. A végső pontosság több futtatás után is 80% körüli értéket ért el, amit kielégítőnek tekintettem, de lehetséges, hogy nagyobb diverzitású tanuló halmazzal (amire kétségkívül szükség lenne), vagy egy ettől eltérő architektúrával jobb eredményt lehetett volna elérni. Érdekes kísérlet lenne különböző architektúrák összevetése, hogy melyiken milyen eredményt sikerül elérnünk ugyanezen az adathalmazon, de ez a felvetés már nem témája ennek a jegyzőkönyvnek.

Hivatkozások

- [1] https://www.diffen.com/difference/Anime_vs_Cartoon
- [2] <https://hu.wikipedia.org/wiki/Anime>
- [3] <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>
- [4] <https://medium.com/datadriveninvestor/convolution-neural-networks-vs-fully-connected-neural-networks-8171a6e86f15>
- [5] <https://www.tensorflow.org/tutorials/images/cnn>
- [6] <https://addons.mozilla.org/hu/firefox/addon/batch-image-downloader/>
- [7] <https://www.youtube.com/watch?v=wQ8BIBpya2k&list=PLQVvva0QuDfhTox0AjmQ6tvTgMBZBEXN>

- [8] <https://datascience.stackexchange.com/questions/24511/why-should-the-data-be-shuffled-for-machine-learning-tasks>
- [9] <https://keras.io/visualization/>
- [10] <https://www.kaggle.com/nabanitaroy/cnn-binary-classification-model>