

Tässä demonstraatiiodokumentissa kuvataan TensorFlow-kirjastolla tehtyjä luokittelukokeita, joissa selvitettiin opetusjoukon koon vaikutusta verkon tarkkuuteen. Ensimmäisessä Laaki-hankkeen TensorFlow-demossa käytettiin opetukseen yli 16 000 kuvaa. Nyt haluttiin selvittää, montako kuvaa tarvitaan, että lopputulos on siedettävä.

TensorFlow on Googlen kehittämä avoimen lähdekoodin tekoäly- ja koneoppimiskirjasto, joka on kenen tahansa ladattavissa Internetistä. Kuvatussa demonstraatioissa käytetään TensorFlow'ta Python-ohjelmointikielellä. TensorFlow'n perusteet kuvattiin edellisessä dokumentissa, joten ne lukea sieltä. Kuvadatanä käytetään Schlagenhauhin (2021) kuvakokoelmaa kunnollisista ja kuluneista kuularuuveista eli samaa kuvasettiä kuin edellisessä TensorFlow-demossa.

Ohjelmointikieli: Python. Saatavilla: <https://www.python.org/>

Kehitysympäristö: Visual Studio Code. Saatavilla: <https://code.visualstudio.com/>

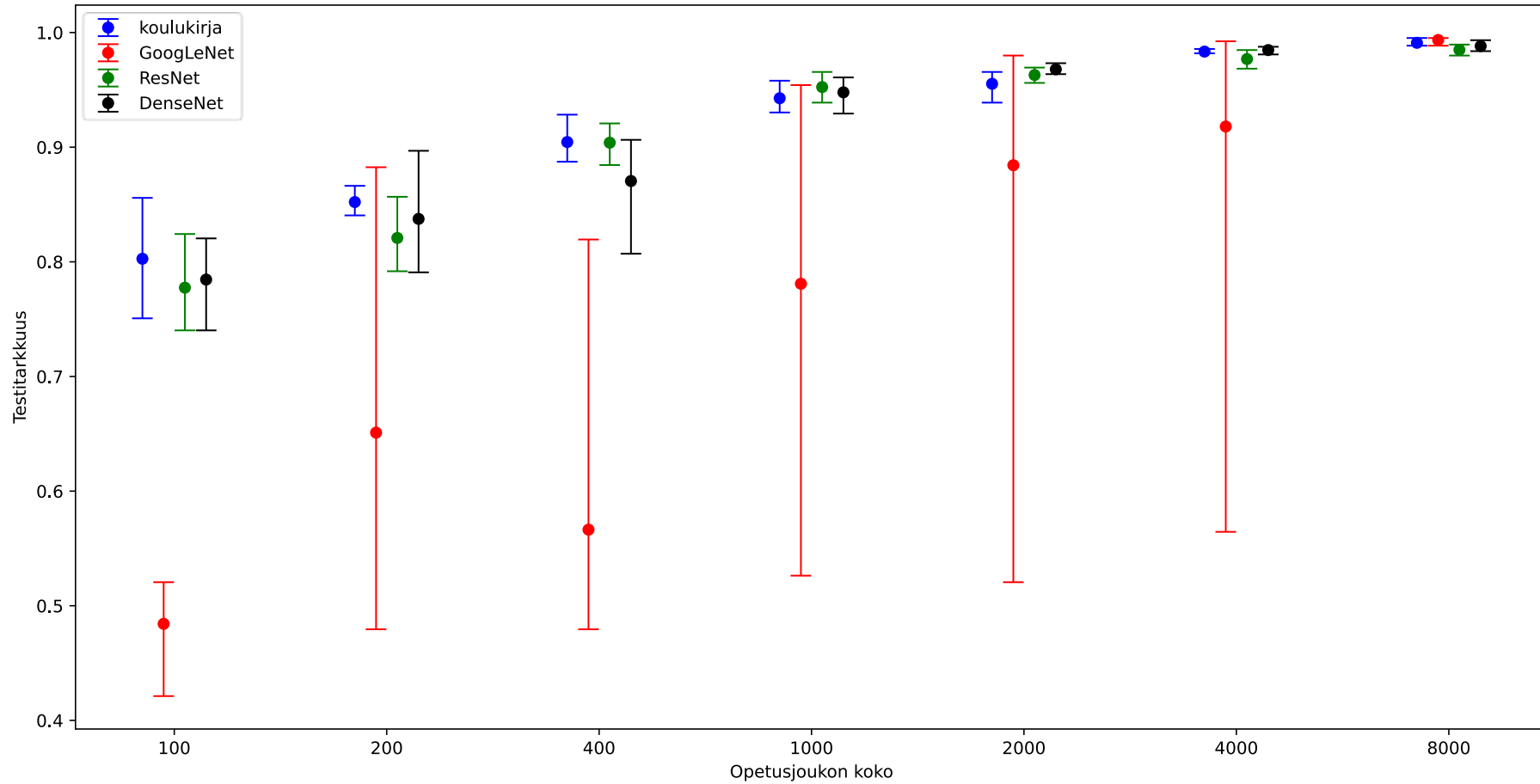
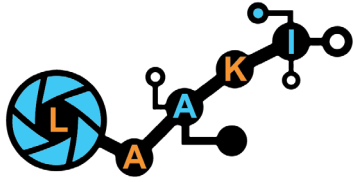
Kirjasto: TensorFlow. Saatavilla: <https://www.tensorflow.org/install>

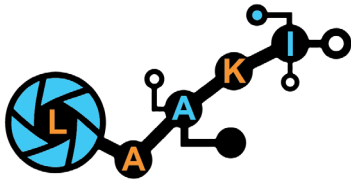
Kuvadata saatavilla: <https://bwdatatdiss.kit.edu/dataset/323>

Käytetystä 16 000 kuvan arkistosta otettiin testeissä satunnaisesti kuvia, ja niistä muodostettiin pienempiä kuvajoukkoja, joita käytettiin mallin opetuksessa. Käytössä olivat koot 100, 200, 400, 1000, 2000, 4000 ja 8000 kuvaa. Puolet kuvista olivat kunnollisista kuularuuveista ja puolet kuluneista. Opetus toistettiin viisi kertaa kullakin koolla eli käytössä oli viisi satunnaistettua kuvajoukkoa jokaista kokoa. Lisäksi kokeiltiin neljää erilaista neuroverkkomallia. Edellisessä dokumentissa kuvatun ns. koulukirjaesimerkin lisäksi kokeiltiin kansainvälisen ImageNet-luokittelukilpailun voittajia eri vuosilta. Käytössä olivat GoogLeNet (Szegedy ym., 2016), ResNet (He ym., 2016) ja DenseNet (Huang ym. 2017). ImageNet-kilpailussa opetusjoukon koko on miljoona kuvaa ja eri luokkia on tuhat. Se on siis paljon monimutkaisempi ongelma kuin tässä testattu kahden luokan ongelma. Kaikki kolme verkkoa rakennettiin Dive into Deep Learning -kirjan (Zhang ym. 2021) ohjeiden mukaisesti.

Luotujen neuroverkkomallien (neljä mallityyppiä, seitsemän kuvajoukkokokoa, viisi toistoa = $4 * 7 * 5 = 140$ mallia) tarkkuus mitattiin 1000 kuvan testijoukon avulla. Siis yksinkertaisesti, moniko ennustus menee oikein testijoukosta. Testijoukon kuvat olivat täysin uusia kaikille malleille. Kaikille malleille käytettiin samaa testijoukkoa.

Seuraavalla sivulla esitetään tulokset kuvaajana. X-akselilla on opetusjoukkojen koot ja y-akselilla mallin tarkkuus. Koska jokaista opetusjoukon kokoa kohti tehtiin viisi eri mallia, kuvaaja esittelee keskimääräisen tarkkuuden (pallo) ja maksimi- ja minimi-tarkkuuden (viivat) kullekin koolle ja mallityypille.





Kuten kuvaajasta nähdään, GoogLeNet suoritus yllättävän huonosti, mutta muuten malleilla ei ollut suurta eroa. GoogLeNet-mallin tapauksessa opetus jäi joskus ns. jumiin eli tarkkuus ei lisääntynyt opetuskierron välillä. Tämän takia joukkoon jäi yleensä yksi todella huono tulos (esim. koot 1000, 2000 ja 4000), joka painoi keskiarvoa alas. Pienemmillä kuvajoukoilla (100, 200, 400) GoogLeNet toimi yleisesti huonosti. Voidaan vetää johtopäätös, että GoogLeNet, ResNet ja DenseNet on tehty etupäässä ongelmiin, missä luokkia on paljon enemmän, eivätkä niiden edut näy vain kahden luokan ongelmassa. Ns. koulukirjaesimerkki toimi siis tässä kahden luokan ongelmassa ihan yhtä hyvin kuin monimutkaisemmat mallit, jotka olivat paljon raskaampia opettaa (esim. 8000 kuvalla koulukirjamallin opetus kesti 3 tuntia, DenseNetin 18,5 tuntia, kun käytössä oli Intel i7-7700HQ CPU @ 2.80GHz, 16 Gt RAM).

Tärkeä tulos kokeessa oli, että kohtalaisen pienellä opetusjoukollakin saadaan jo kohtalaisen hyviä tuloksia. Jo 400 kuvaa riitti noin 90 % tarkkuuden saavuttamiseksi. 100 kuvallakin päästiin jo 80 % tarkkuuden tuntumaan. Tämä tarkoittaa, että jo pienellä määrällä kuvia voi saavuttaa hyötyä, jos mallia käytetään esim. antamaan hälytys mahdollisesta vikatilanteesta ihmiselle, ja ihminen käy vielä varmistamassa tilanteen.

Demonstraation tuloksena Seamkilla on käytössä lisäksi nyt koodi, jonka avulla

- 1) voi opettaa GoogLeNet-, ResNet- ja DenseNet-mallit
- 2) voi kokeilla nopeasti opetusjoukon koon vaikutusta mallin oppimiseen.

Lähteet

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition, 770–778.

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. Proceedings of the IEEE conference on computer vision and pattern recognition, 4700–4708.

Schlagenhauf, T. 2021 Ball Screw Drive Surface Defect Dataset for Classification. [Sähköinen tietoaineisto]. Karsruher Institut für Technologie. [Viitattu: 3.9.2021]. Saatavilla: <https://bwdatadiss.kit.edu/dataset/323>, doi: 10.5445/IR/1000133819

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... Rabinovich, A. (2015). Going deeper with convolutions. Proceedings of the IEEE conference on computer vision and pattern recognition, 1–9.)

Zhang, A., Lipton, Z. C., Li, M. & Smola, A. J. 2021. Dive into Deep Learning. [Verkkokirja], arXiv preprint. [Viitattu: 3.9.2021]. Saatavilla: <https://d2l.ai/>, doi: arXiv:2106.11342