



Tekoäly – Me, myself & AI

Työpajakokonaisuuksia
peruskouluun ja toiselle asteelle

Heureka

Oivaltamisen iloa!

Sisältö

Oppimateriaali on suunniteltu tukemaan Heureka Tekoäly – Me, Myself & AI -näyttelyn teemojen käsittelyä kouluissa. Näyttelyssä kävijä pääsee tutustumaan hausalla ja vuorovaikutteisella tavalla tekoälyn kehittämisen perusteisiin, ihmisen ja tekoälyn ohjaamien robottien vuorovaikutukseen sekä tekoälyn kehityksen yhteiskunnallisiin ulottuvuuksiin.

Materiaali koostuu kahdesta työpajasta ja niitä tukevista tehtävistä. Taitava tekoäly - työpaja soveltuu peruskoulun kaikille luokka-asteille ja Tekoälylajittelija Arduinoilla -työpaja peruskoulun yläluokille sekä toiselle asteelle. Peruskouluun suunniteltuja tehtäviä voi käyttää soveltuvin osin myös toisella asteella. Työpajat sopivat tehtäväksi ennen tai jälkeen näyttelykäynnin. Materiaalia voi hyödyntää myös ilman näyttelykäyntiä.



Sisällysluettelo

LYHYESTI TEKÖÄLYSTÄ	4	TEKOÄLYLAJITTELIJA ARDUINOLLA (75-90 MIN)	29
TAITAVA TEKÖÄLY (45-90 MIN)	8	Ohjelman tavoite	30
Ohjelman tavoite	9	Ohjeita materiaalin käyttöön	31
Yhteys perusopetuksen opetussuunnitelmaan	9	Tekoälylajittelija Arduinolla -työpajan vaiheet	32
Ohjeita materiaalin käyttöön	10	Huomioita työpajan eri vaiheista	33
Ohjelman ennakoivalmistelut	10	Johdanto	34
Työpajan vaiheet	10	Tekoälylajittelija Arduinolla -työohjeet	35
Oman tekoälymallin luominen	11	1 Oman tekoälymallin luominen	36
Ohjelmointiprojektin toteuttaminen: AI Blocks	11	2 Lajittelijailmaisimen rakentaminen led-valoista	39
Ohjelmointiprojektin toteuttaminen: MakeCode	12	3 Ohjelmakoodin valmistelu	40
Jekuta kuvantunnistusta	12	4 Arduinon yhdistäminen tekoälymalliin	42
Lisämateriaalia opettajalle ohjelmoinnin tueksi	13	5 Tekoälylajittelijan testaaminen	42
Tehtäviä Taitava tekoäly -työpajaohjelman tueksi	14	Lähteet ja lisämateriaalia	43
Tehtävä: Arvaa tekoäly, mitä piirrän	15	Tehtäviä Tekoälylajittelija Arduinolla -työpajan tueksi	44
Tehtävä: Testaa lukeeko tekoäly ajatuksesi	16	Tehtävä: Oman mekaanisen tekoälyn rakentaminen	45
Tehtävä: Taiteile tekoälyn kanssa	17	Tehtävä: Mitä YouTube oppii sinusta?	47
Tehtävä: Pystyykö tekoäly huijaamaan sinua?	18	LISÄTIETOA	51
Oppimateriaalin ohjelmien käyttöohjeet	20	Koneoppiminen	52
Ohjeet: Teachable Machine	21	Tekoälyn etiikka	53
Ohjeet: AI Blocks	24	Supertietokoneet tekoälyn kehityksen taustalla	54
Ohjeet: Microbit of AI	28		

Lyhyesti tekoälystä

Termit tutuiksi

Tekoäly

Tietokoneohjelma, joka jäljittelee inhimillistä älykkyyttä vaativaa toimintaa kuten päättelystä, kuvien tunnistamista tai puhetta.

Koneoppiminen

Toiminta, jossa tietokone tarkastelee sille annettua opetusaineistoa ja tuottaa siitä mallin, jolla se voi tehdä päätelmiä.

Opetusaineisto

Esimerkkejä sisältävä aineisto eli data, jolla koneoppimiseen perustuvalla tekoälyllä opetetaan haluttuja asioita.

Neuroverkko

Keinotekoinen hermoverkko, joka jäljittelee toiminnassaan biologisia hermoverkkoja. Neuroverkko koostuu neuroneista ja niiden välisistä liitoksista eli keinotekoisista synapseista. Neuroverkkoja hyödynnetään koneoppimisessa.

Neuroni

Neuroverkon pienin tiedonkäsittely-yksikkö, joka tallentaa ja prosessoi tietoa. Neuronit lähettävät toisilleen viestejä ja muodostavat muutettavat mallit opetusaineistolle.

Algoritmi

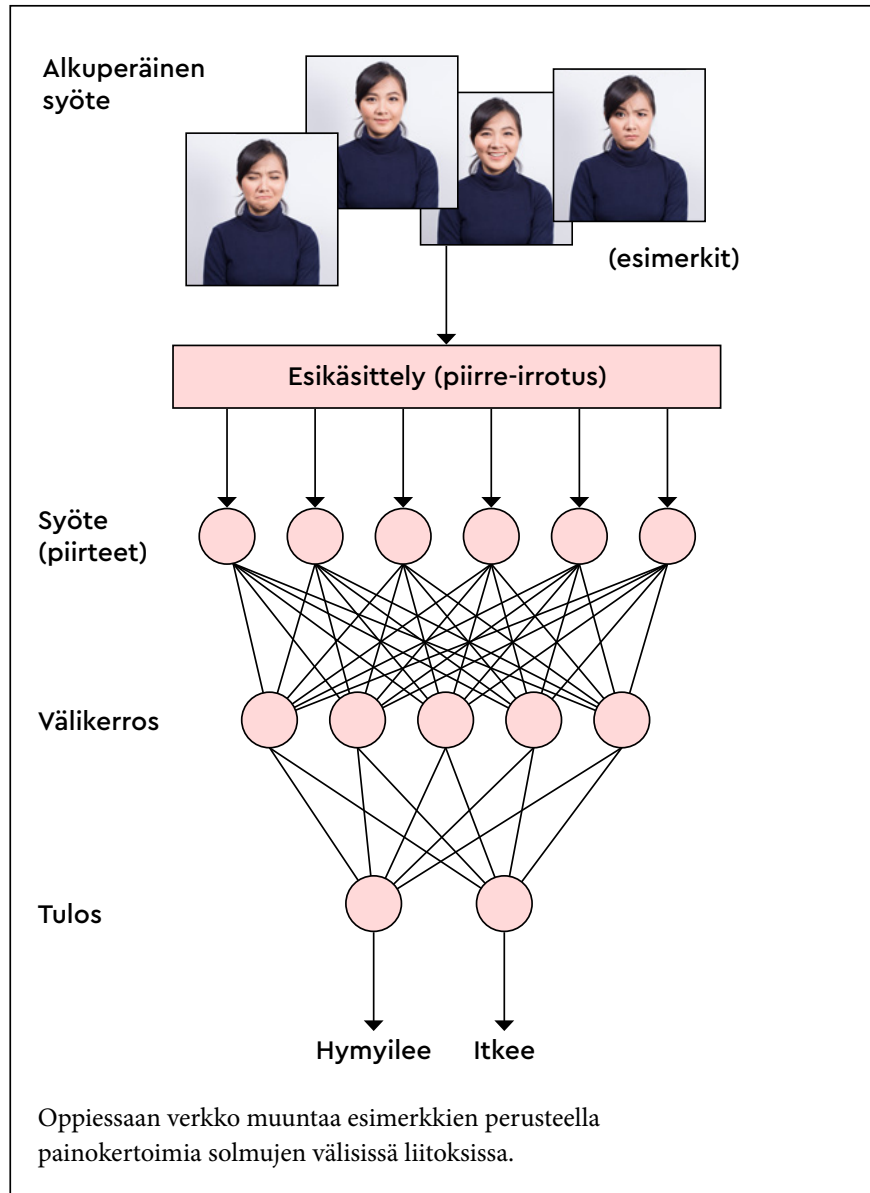
Tarkka kuvaus askeleista, joita seuraamalla tietokone voi suorittaa tietyn tehtävän tai ratkaista tietyn ongelman.

Tekoälyllä ei ole yhtä yleisesti hyväksyttyä määritelmää. Kaplanin ja Haenleinin (2019) määritelmän mukaan tekoäly voidaan ymmärtää "järjestelmän kyvyksi tulkitella ulkoisia tietoja oikein, oppia tällaisista tiedoista ja käyttää opittuja asioita tiettyjen tavoitteiden ja tehtävien saavuttamisessa joustavan sopeutumisen kautta." Tekoälyä voidaan tarkastella myös sen ominaisuuksien kautta. Tyypillisesti tekoälyllä on kyky suoriutua tehtävistä monimutkaisissa ympäristöissä ilman jatkuvaa ihmisen suoritamaa ohjausta. Se pystyy parantamaan suoritustaan oppimalla kokemuksesta.

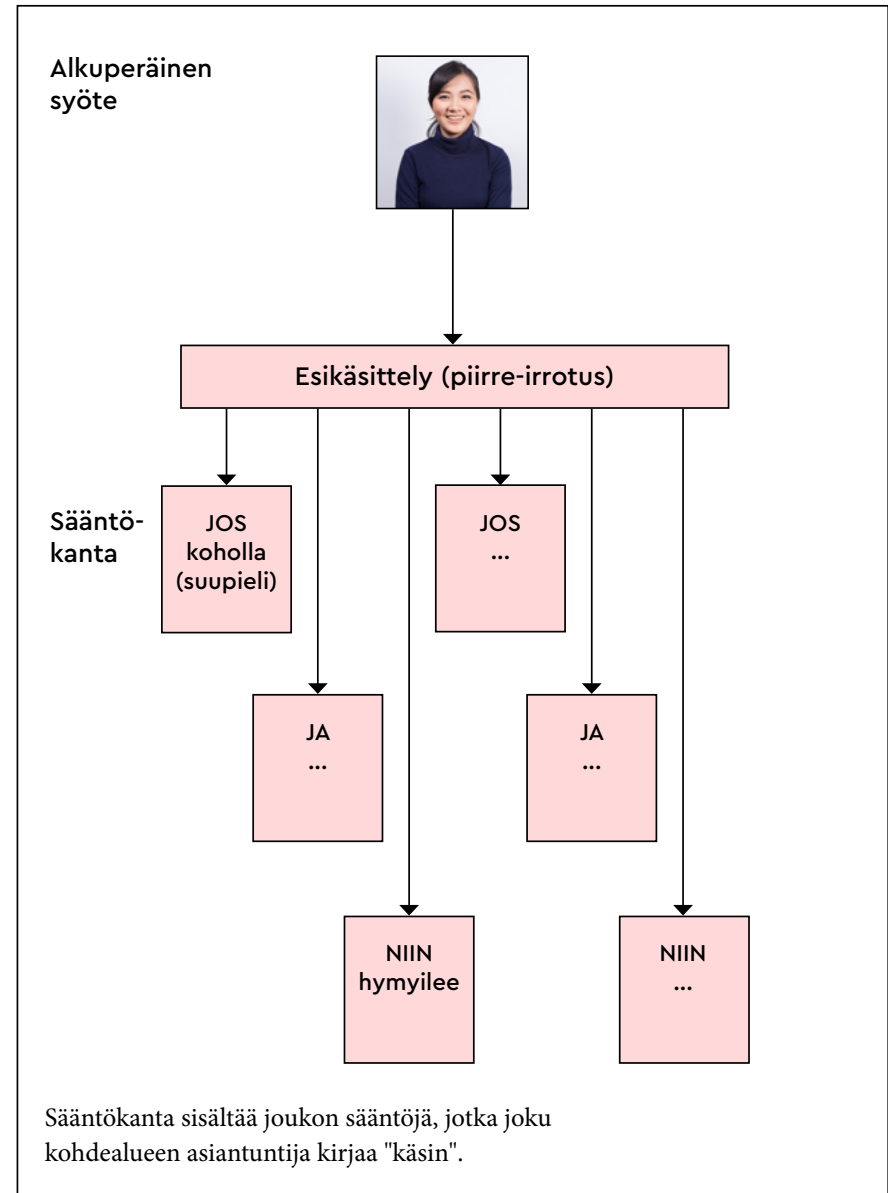
Kumman luulet olevan tekoälyjärjestelmän näkökulmasta haastavampaa, shakin pelaamisen vai kynän käteen ottamisen? Jos vastasit kynän ottamisen käteen, onneksi olkoon! Olet edelläkävijä tekoälyn ymmärtämisen saralla. Tekoälytutkimus keskittyi alkuaikoina ratkaisemaan juuri esimerkiksi shakin pelaamiseen tai matemaattisiin pulmiin liittyviä tehtäviä. Shakkia pelaava supertietokone Deep Blue luotiin jo 1990-luvulla, mutta esineisiin robottikädellä tarttumiseen liittyviä haasteita ratkotaan yhä. (Tässä yksi esimerkki Googlen robottikädestä: <https://spectrum.ieee.org/google-large-scale-robotic-grasping-project>.)

Deep Blue oli sääntöpohjainen tekoäly eli se noudatti ihmisen kirjoittamia toimintaohjeita. Sen menestys perustui algoritmeihin, joiden avulla Deep Blue pystyi laskemaan shakkipeliä jopa satoja siirtoja eteenpäin. Nykyään käytetään neuroverkko-pohjaisia tekoälysovelluksia, jotka eivät olisi vielä 1990-luvulla olleet mahdollisia tietokoneiden rajallisen laskentatehon ja muistin vuoksi. Neuroverkot jäljittelevät rakenteeltaan aivojen hermoverkkoja. Ne oppivat esimerkkien pohjalta ja neuroverkkojen toiminta perustuu opetusaineistojen käyttöön.

Neuroverkkoratkaisu



Sääntöpohjainen lähestymistapa



Tämän oppimateriaalin työpajoissa tekoälymallia koulutetaan tunnistamaan esineitä. Kuvantunnistuksessa tekoäly tunnistaa esineitä niiden ominaisuuksien perusteella. Mitä enemmän ja monipuolisemmin tekoälymallia harjoittaa, sitä monimutkaisempia ominaisuuksia se pystyy tunnistamaan. Otetaan esimerkkinä yksinkertaistettu havainnollistus siitä, kuinka tekoäly oppii tunnistamaan kalan:

Päätös 1: Hahmottaa (suomun) tumman ja vaalean reunan

Päätös 2: Tunnistaa näiden tummien ja vaaleiden alueiden muodostavan kaaren

Päätös 3: Tunnistaa niiden muodostavan suomun

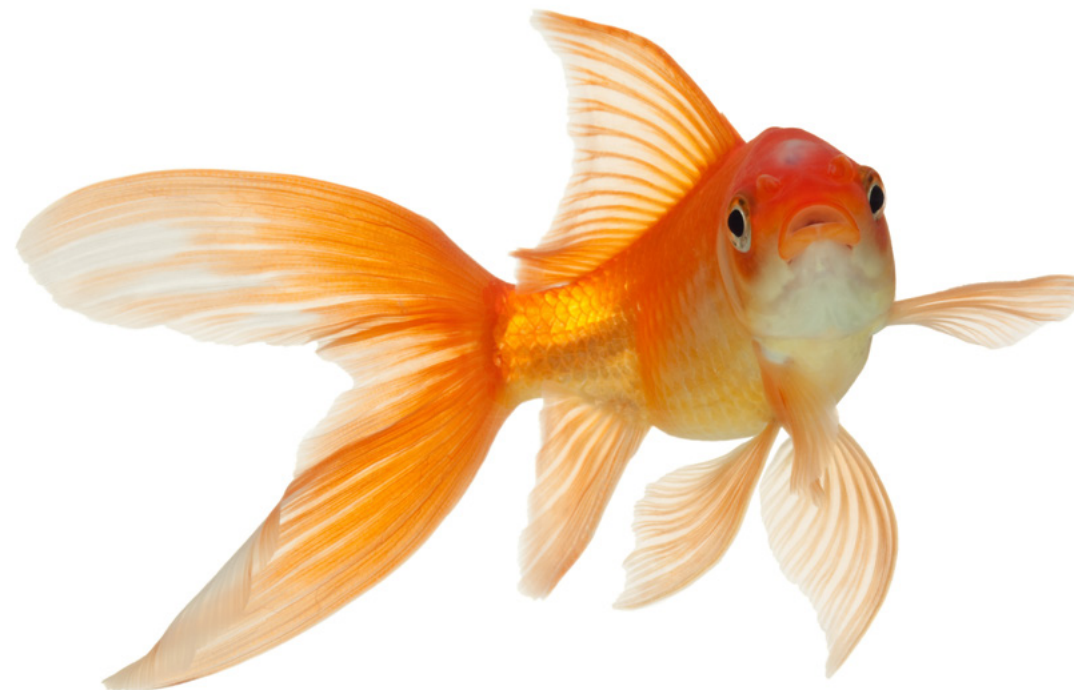
Päätös 4: Tunnistaa suomuista muodostuvan vartalon (sama prosessi tapahtuu myös muille kalalle tyypillisten ominaisuuksien kohdalla, kuten evä tai silmä)

Päätös 5: Ryhmittelee aiemmissa vaiheissa tehdyt pienemmät päätökset yhteen ja pääättelee kuvassa olevan kala.

Tekoälyn kuvantunnistus ja -käsittely ovat edenneet suurin harppauksin viimeisen 10 vuoden aikana. Monelle on tuttu Deepfake-menetelmä, jossa kenen tahansa kasvot voidaan vaihtaa videokuvaan. Se perustuu neuroverkkojen Generative Adaptive Networks (GAN)-koulutusmenetelmään. GAN-menetelmässä kaksi neuroverkkoa kouluttavat toisiaan kilpailemalla toisiaan vastaan. Esimerkiksi Deepfake-kuvia tehdessä generatiivinen neuroverkko tuottaa uusia kuvia opetusaineiston pohjalta ja toinen eli kilpaileva neuroverkko yrittää tunnistaa tekoälyn tuottamat kuvat oikeista.

Uusinta neuroverkkoteknologiaa edustavat transformer-verkot. Ne pystyvät muis-tamaan aiemmin syötettyä sisältöä ja kiinnittämään huomiota saamansa syötteen kontekstiin. Tätä teknologiaa on näiden ominaisuuksien vuoksi hyödynnetty paljon tekstipohjaisissa sovelluksissa, kuten chatboteissa ja automaattisissa tekstin kää-nösohjelmassa.

Tekoäly voidaan jakaa vahvaan ja heikkoon tekoälyyn. Heikko tekoäly tarkoittaa yksittäisiin tehtäviin suunniteltuja tekoälyjärjestelmiä, jotka kykenevät tekemään vain ennakkoon niille ohjelmoituja tehtäviä. Kaikki tämänhetkiset tekoälysovellukset kuuluvat heikkoon tekoälyyn. Vahva tekoäly pystyisi puolestaan toimimaan täysin itsenäisesti ilman ihmisen ohjausta. Tätä fiktiivisissä tieteiselokuvissa joskus nähtyä inhimillistä älykkyyttä vastaavaa tekoälyä ei ole vielä pystytty kehittämään.



Lähteet ja lisämateriaalia

Helsingin yliopisto & MinnaLearn. Elements of AI. <https://course.elementsofai.com/fi/>

Helsingin yliopisto & MinnaLearn. Building AI. <https://buildingai.elementsofai.com/>

Jyväskylän yliopisto. Tekoälyn perusteita ja sovelluksia. <https://tim.jyu.fi/view/kurssit/tie/tiep1000/tekoalyn-sovellukset/kirja#DKUvbnUuGytQ>

Kaplan A. & Hainlein M. (2019). Siri, Siri, in my hand: Who's the fairest in the land? On the interpretations, illustrations, and implications of artificial intelligence. <https://www.sciencedirect.com/science/article/abs/pii/S0007681318301393>

Ojala, P. (2022). Neuroverkkopohjaiset tekoälysovellukset: kehityssuuntia ja tulevaisuuden mahdollisuuksia. <https://karelia.fi/2022/08/neuroverkkopohjaiset-tekoalysovellukset-kehityssuuntia-ja-tulevaisuuden-mahdollisuuksia/>

Steamlabs. Visual Recognition. <https://steamlabs.ca/programs/vision-recognition/>

Steamlabs. Visual Recognition. Google Slides. (CC BY-SA 4.0) https://docs.google.com/presentation/d/1LiRYqBcn6lPgrXDIVKzymmnuqJmL08IowRXhdexWBZg/edit#slide=id.gcba4964ea2_0_13

Taitava tekoäly (45–90 min)

Ohjelma on suunniteltu tukemaan Heureka Tekoäly – Me, Myself & AI -näyttelyn sisältöjä, ja työpaja soveltuu niin ala- kuin yläkouluun.

Ohjelman tavoite

Tavoitteena ohjelmassa on tutustua tekoälyn ja koneoppimisen käsitteisiin ja erityisesti kuvantunnistukseen.

Yhteys perusopetuksen opetussuunnitelmaan

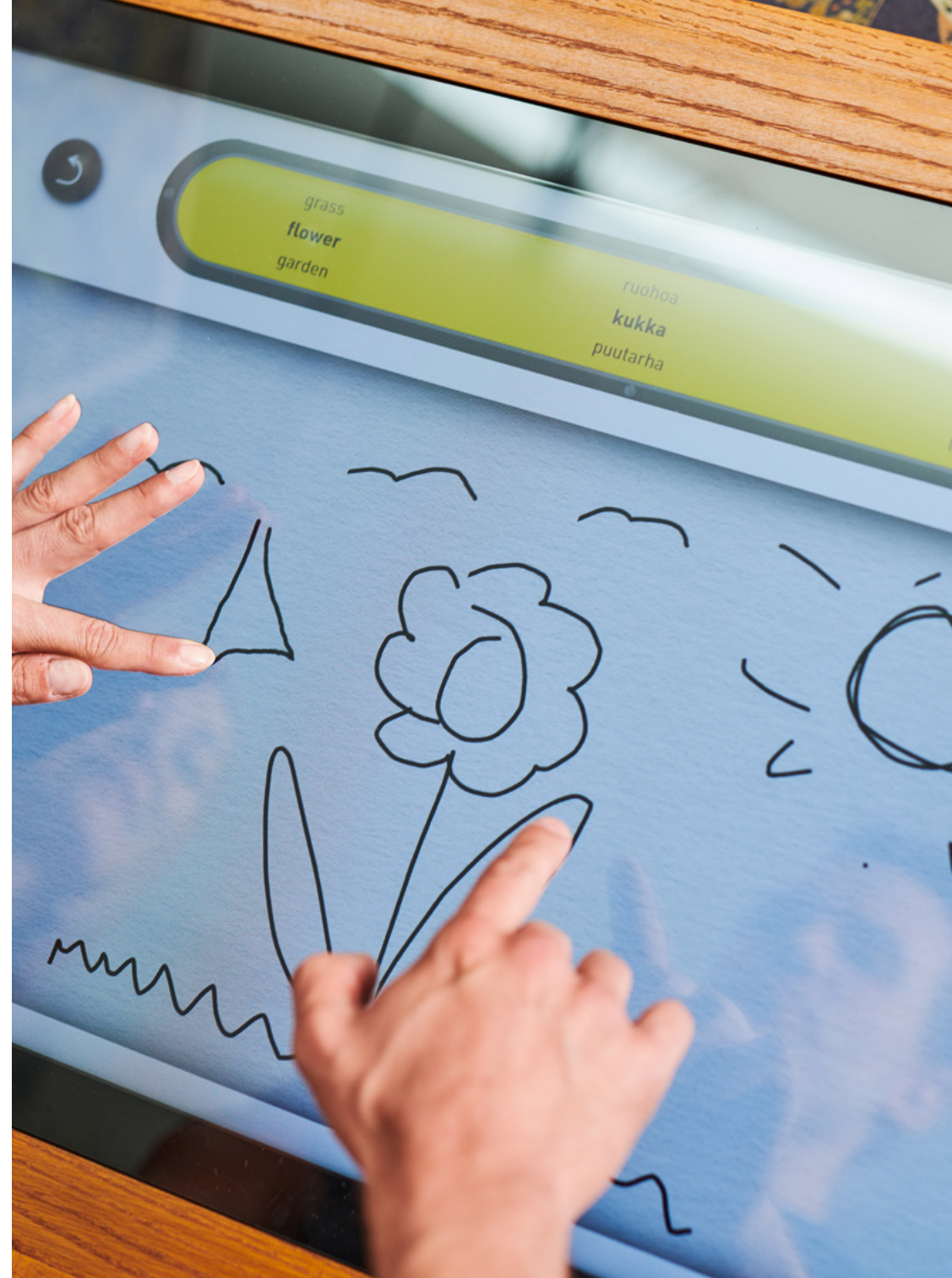
Alakoulun opetussuunnitelmassa (2014) tekoälyyn ja tarkemmin koneoppimiseen tutustuminen liittyy tieto- ja viestintäteknologisen osaamisen laaja-alaisiin tavoitteisiin (L5). Vaikkei tekoälyä mainita erikseen, ohjelmointia harjoitellaan eri oppiaineiden opinnoissa (mm. matematiikassa ja käsityössä). Tarkempi erittely opetussuunnitelman tekoälyyn ja ohjelmointiin liittyvistä tavoitteista vuosiluokittain:

Vuosiluokilla 1-2

- Laaja-alainen osaaminen (L5): Oppilaat saavat ja jakavat keskenään kokemuksia digitaalisen median parissa työskentelystä sekä ikäkaudelle sopivasta ohjelmoinnista (OPS 2016, 101)
- Matematiikka: Tutustuminen ohjelmoinnin alkeisiin alkaa laatimalla vaihteellaisia toimintaohjeita, joita myös testataan (OPS 2016, 129)

Vuosiluokilla 3-6

- Laaja-alainen osaaminen (L5): Ohjelmointia kokeillessaan oppilaat saavat kokemuksia siitä, miten teknologian toiminta riippuu ihmisen tekemistä ratkaisuista (OPS 2016, 157).
- Matematiikka: T14 innostaa oppilasta laatimaan toimintaohjeita tietokoneohjelmoinnissa graafisessa ohjelmointiympäristössä (OPS 2016, 235).
- Käsityö: Harjoitellaan ohjelmoimalla aikaan saatuja toimintoja, joista esimerkiksi robotiikka ja automaatio (OPS 2016, 271).



Ohjeita materiaalin käyttöön

Työpajan kesto on 45–90 min ja soveltuu käytettäväksi myös ilman näyttelykäyntiä. Työpajaohjelman tueksi on Powerpoint-esitys.

Ohjelma aloitetaan johdattelemalla osallistujia tunnistamaan tekoälyä hyödyntäviä laitteita omasta arjesta. Tämän jälkeen tutustutaan koneoppimisen periaatteisiin kuvantunnistuksen kautta.

Työpajan voi tehdä kokonaan, tai se osioista voi tehdä vain osan. Työpajan ensimmäisessä osassa luodaan oma tekoälymalli [Googlen Teachable Machine](#) -työkalun avulla. Toisessa osassa itse luotua mallia hyödynnetään oman ohjelmointiprojektin tekemiseen. Ohjelmointiprojektin voi toteuttaa Scratch-pohjaisessa [AI Blocks](#) -ohjelmointiympäristössä tai [MakeCode](#)-editorissa, jos esimerkiksi haluaa käyttää projektissa BBC Microbit -mikrokontrolleria. Osiot Oman ohjelmointiohjelman tekeminen ja Jekuta kuvan tunnistusta voi tehdä joko molemmat tai valita vain toisen.

Keskeiset käsitteet:

tekoäly, koneoppiminen, ohjattu oppiminen

Ohjelman ennakkovalmistelut

Valmisteluihin kannattaa varata noin 15–20 minuuttia. Tietokoneet kannattaa käynnistää ja avata niihin Teachable Machine -sivusto. Ohjelmointiprojektia varten haluttu ohjelmointialusta (AI Blocks tai MakeCode) kannattaa lisätä kirjanmerkkipalkkiin kirjanmerkiksi.

Tarvikkeet:

- Tietokoneet/tabletit
- Erilaisia lajiteltavia esineitä:
 - eläinleluja
 - kyniä
 - hedelmiä yms.

Tietokoneella lisäksi tallennettuna:

- Google Teachable Machine: <https://teachablemachine.withgoogle.com/>
- AI Blocks: <https://mitmedialab.github.io/prg-extension-boilerplate/create/> tai
- MakeCode: <https://makecode.microbit.org/> ja
Microbit of AI: <https://ai-training.glitch.me/>

Työpajan vaiheet

Johdanto

Työpaja aloitetaan tutustumalla siihen, missä kaikkialla hyödynnämme tekoälyä arkielämässä. Diassa 2 on kuvia erilaisista arjessa käytettävistä asioista. Jaa oppilaat pareihin tai pieniin ryhmiin. Heidän tehtävänä on luokitella asiat sen mukaan, käyttävätkö ne tekoälyä vai ei.

Seuraavaksi jatketaan johdattelua antamalla oppilaille mahdollisuus asettaa itse tekoälyn asemaan. Dioissa 3–13 on kuvia, jotka oppilaiden tulee luokitella joko kissaksi tai koiraksi. Dioista voi näyttää haluamansa ja piilottaa loput. Tarkoituksena saada osallistujat havaitsemaan, ettei taitoa tunnistaa ja luokitella kuvia ole aivan niin yksinkertaista opettaa koneelle.

Ihmisillä on tuhansien toistojen tuoma varmuus, jolla voimme tunnistaa esimerkiksi kissan jo hyvin pienistä vihjeistä kuvassa. Olemme alitajuisesti oppineet, mikä tekee kissasta kissan. Tekoäly ei opi tätä itsestään, vaan ihmisen tulee opettaa sille se. Tekoälyä opetettaessa on tärkeää kiinnittää huomiota siihen, millaisia kuvia valitsemme opetusmateriaaliksi. Valitut kuvat ratkaisevat, oppiiko tekoäly tunnistamaan kissan kissana vai kenties taustalla olevan jalkakäytävän.

Oman tekoälymallin luominen

Välineet:

Tietokoneet, lajiteltavia esineitä, Teachable Machine -ohje

Osion tavoite:

Luoda oma tekoälymalli

Osiossa luodaan oma tekoälymalli [Googlen Teachable Machine](#) -työkalun avulla. Oppimateriaalissa annetaan ohjeet kuvantunnistukseen, mutta projektin voi toteuttaa myös hyödyntäen äänen tai eleiden tunnistusta.

Tekoälymallin luomisessa on kolme vaihetta: 1) opetusaineiston kerääminen, 2) tekoälymallin kouluttaminen ja 3) mallin lataaminen. Opetusaineiston keräämisessä oppilaat valitsevat ensin esineet, mitä haluavat tekoälymallin tunnistavan (esim. pallo ja kynä). He luovat Teachable Machine -työkaluun jokaiselle tunnistettavalle esineelle oman luokan (esim. Pallo) ja keräävät opetusaineiston ottamalla esineistä kuvia tietokoneen kameralla. Kun kuvia on riittävästi (vähintään 40 kuvaa / luokka), tekoälymalli koulutetaan kerätyllä opetusaineistolla. Nyt mallia voi testata Teachable Machine -työkalussa ja tarvittaessa lisätä tai poistaa kuvia opetusaineistosta.

Käy läpi lyhyesti Teachable Machinen toiminnot. Halutessasi voit jakaa ryhmille ohjeet tulosteina. Kannattaa erityisesti nostaa esille ohjeista seuraavat seikat:

- Taustaluokan luominen parantaa oman mallin tarkkuutta
- Luokan sisällä kannattaa olla monipuolisesti erilaisia sitä luokkaa edustavia kuvia

Kun tekoälymalli on valmis, sen voi siirtää ohjelmointiprojektiin käyttämällä Shareable link -toimintoa. AI Blocks -ohjelmointiympäristöön tekoälymallin voi tuoda suoraan, mutta Makecode -ohjelmointiympäristöä varten tarvittavat lisäksi Microbit of AI -sivuston.

Ohjelmointiprojektin toteuttaminen: AI Blocks

Välineet:

Tietokoneet, AI Blocks template-pohja, AI Blocks -ohje

Osion tavoite:

Hyödyntää omaa tekoälymallia ohjelmoinnissa

Tässä osiossa itse luotua mallia hyödynnetään Scratch-pohjaisessa **AI Blocks** -ohjelmointiympäristössä oman ohjelmointiprojektin tekemiseen. Ryhmäsi mukaan käy joko yhdessä läpi AI Blocks -ohjelmointisivuston käyttö tai anna osallistujien siirtyä joustavasti itsenäisesti eteenpäin. Template-pohjaan on kerätty tyypillisimpiä tarvittavia ohjelmointipalikoita. Template-pohjaa ei välttämättä tarvita, jos graafinen ohjelmointi on osallistujille jo tuttu.

Perusesineiden tai esimerkiksi ilmeiden tunnistuksen lisäksi muutama lisätehtävä ohjelmointiin:

- kivi, paperi, sakset -peli
- hahmon ohjaaminen kuvien avulla

Eriyttämistehtävänä oppilaat voivat tehdä kuvantunnistuksen lisäksi äänen tai asennon tunnistamiseen perustuvan ohjelman.

Ohjelmointiprojektin toteuttaminen: MakeCode

Välineet:

Tietokoneet, Microbit-mikroprosessorit, halutut lisätarvikkeet (esim. servomootoreita, johtoja, pahvia, saksia, kyniä)

Osion tavoite:

Hyödyntää omaa tekoälymallia ohjelmoinnissa

Tässä osiossa itse luotua tekoälymallia hyödynnetään MakeCode-ohjelmointiympäristössä oman ohjelmointiprojektin tekemiseen. Tekoälymalli tuodaan Makecode-ohjelmointiympäristöön Microbit of AI -sivuston avulla.

Osio aloitetaan ohjelmoimalla haluttu tekoälymallia hyödyntävä ohjelma. Peruselementtien tai esimerkiksi ilmeiden tunnistuksen lisäksi tässä on muutama lisäidea ohjelmointiin:

- kivi, paperi, sakset -peli
- lajitteleva laite
- tekoälyä hyödyntävä auto

Ohjelmointipohjaan on kerätty tärkeimmät ohjelmointipalikat, joiden avulla saa siirrettyä tekoälymallin Makecode-ohjelmointiympäristöön. Tekoälymalli on tehty Teachable Machine -työkalulla. Pohjassa on myös yksinkertainen esimerkki If-lauseohjelmoinnista, jolla saa Microbitin vaihtamaan palavien ledien järjestystä luokiteltavan esineen mukaan.

Ohjelmoinnin jälkeen suoritetaan Microbitin yhdistäminen tekoälymalliin. Siinä käytetään apuna Microbit of AI -sivustoa. Erilliset ohjeet löytyvät työpajan liitteistä.

Eriyttämistehtävänä oppilaat voivat tehdä ohjelman kuvantunnistuksen lisäksi äänen tai asennon tunnistamiseen perustuvan ohjelman.

Jekuta kuvantunnistusta

Välineet:

Tietokone, lajiteltavia esineitä

Osion tavoite:

Oppia, millaiset tekijät vaikuttavat tekoälymallin luotettavuuteen

Tässä osiossa tarkoituksena on yrittää huijata aiemmin tehtyä tekoälymallia eli estää ohjelmaa tunnistamasta opetettuja esineitä. Tekoälytutkijat etsivät koko ajan kehittämiensä tekoälymallien rajoja eli yrittävät jekuttaa konetta. Tämän jekuttamisen tarkoituksena on kehittää tekoälymalleista luotettavampia. Esimerkiksi itseohjautuvan auton on tärkeä pystyä tunnistamaan liikennemerkkejä, vaikka ne olisivat osittain lumen peitossa.

Osion voi tehdä suoraan käyttämällä vain Teachable Machine -mallia tai yhdistämällä siihen ohjelmointia.

Oppilaat voivat myös halutessaan vaihtaa malleja toisen ryhmän kanssa ja yrittää jekuttaa toisten mallia.

Lopetus

Lopuksi kootaan yhteen työpajan aikana opittua. Koontikysymyksinä voi esimerkiksi käyttää:

- Mitä on tekoäly?
- Mitä tulee ottaa huomioon valittaessa tekoälyn opetusaineistoa?
- Mihin voisimme nyt ja tulevaisuudessa hyödyntää tekoälypohjaista kuvantunnistusta? Esimerkiksi:
 - Tekstistä puheeksi -sovellukset
 - Lisätty todellisuus (AR)
 - Itseohjautuvat autot

Lähde ja lisätieto: <https://www.datasciencesociety.net/image-recognition-applications-7-essential-future-uses/>

Lisämateriaalia opettajalle ohjelmoinnin tueksi

- Innokas-verkosto: <https://www.innokas.fi/>
- Kansallinen audiovisuaalinen instituutti. Polkuja ohjelmointiosaamiseen. Opas vuosiluokille 1–6: <https://uudetlukutaidot.fi/tuki-ja-julkaisut/polku-ja-ohjelmointiosaamiseen-opas-vuosiluokille-1-6/>
- Kansallinen audiovisuaalinen instituutti. Polkuja ohjelmointiosaamiseen. Opas vuosiluokille 7–9: <https://uudetlukutaidot.fi/tuki-ja-julkaisut/polku-ja-ohjelmointiosaamiseen-opas-vuosiluokille-7-9/>
- Helsingin yliopiston Tiedeluokka Linkin materiaalipankki: <https://linkki.cs.helsinki.fi/fi/materials/>
- Microbit-ohjelmointiprojekteja: <https://microbit.org/projects/>
- Vinkkejä ilmiöpohjaisiin ohjelmointiprojekteihin: <https://mehackit.org/kurssit/>



Tehtäviä Taitava tekoäly -työpajaohjelman tueksi



Tehtävä

Arvaa tekoäly, mitä piirrän

Koneoppimisella voi opettaa tekoälyä kuvantunnistuksen lisäksi jatkamaan ihmisen tekemää piirrosta. Tehtävässä oppilaat pääsevät vertailemaan tekoälysovelluksen ja ihmisen kykyä arvata tekeillä oleva piirros.

Piirrä & arvaa

- Kokeile ensin parin kanssa pelata hetki Piirrä & arvaa -peliä. Piirrä kuvat ensin mahdollisimman helposti tunnistettavasti. Tämän jälkeen yritä jekuttaa paria. Saatto hänet olemaan arvaamatta kuvaa?
- Pelaa sitten samaa peliä tekoälysovelluksen kanssa. Voit pelata joko Heureka näyttelyssä tai sitten Googlen [Quick, Draw!](#) -sivustolla (sivusto englanninkielinen).
- Kumpi tunnisti nopeammin, mitä aiot piirtää: tekoäly vai ihminen? Mistä luulet sen johtuvan?

Piirrä & jatka

- Kokeile pelata parin kanssa Piirrä & jatka -peliä. Kerro ensin parille, mitä aiot piirtää ja aloita piirros. Lopeta piirtäminen haluamassasi kohdassa ja anna piirros parillesi. Hän jatkaa piirroksen loppuun. Kokeile:
 - Vaikuttaako parisi arvaukseen, kuinka valmiiksi piirrät kuvasi.
 - Älä kerro parillesi, mitä piirrät. Onnistuuko parisi jatkamaan kuvan oikein loppuun?
- Kokeile nyt samaa [Magic Sketchpad](#) -tekoälysovelluksen kanssa (sivusto englanninkielinen).
- Kumpi piirsi mielestäsi aloittamasi kuvan paremmin loppuun: tekoäly vai ihminen? Vaikuttiko suoritukseen, kuinka pitkälle piirsit kuvan ennen sen antamista tekoälylle tai parillesi?



Tehtävä

Testaa lukeeko tekoäly ajatuksesi

Tekoälysovellukset voivat oppia ymmärtämään puhuttua tai kirjoitettua kieltä luonnollisen kielen käsittelyyn erikoistuneiden Natural Language Processing (NLP) -koneoppimismallien avulla. Tässä tehtävässä testataan, kuinka hyvin tekoäly ymmärtää sinua:

- a. Osaako tekoäly jatkaa lauseesi loppuun?
Kokeile: <https://ajatussampo.azurewebsites.net/>

- b. Tekoäly yrittää tulkita tunnetilaasi: <https://tunnekone.azurewebsites.net>
Kokeile:
 - Testaa, miten tekstisi pituus vaikuttaa lopputulokseen. Saatko vaihdettua tekoälyn ehdotuksia lisäämällä tai poistamalla sanoja?
 - Kokeile poistaa eri sanaluokkiin kuuluvia sanoja. Miten niiden poistaminen vaikuttaa? Onko jollain sanaluokalla suurempi vaikutus tekstiin?
 - Kokeile myös käyttää erilaisia tekstityylejä. Muuttuvatko tulokset? Miten?



Tehtävä

Taiteile tekoälyn kanssa

Yksi tekoälytutkimuksen kohde on tekoälyn luovuus. Transformer-neuroverkoilla on tehty niin runoutta kuin taidettakin yhdessä ihmisen kanssa. Tässä tehtävässä testataan tekoälyn luovuutta:

- a. Craiyon-tekoälysovellus muodostaa kuvan kirjoittamasi tekstin perusteella. (Huom. Sovellus ymmärtää ainoastaan englantia.) Voit kertoa sille sekä, mitä haluat kuvassa olevan, että millä tyylillä haluat sen tekevän kuvan. Esimerkiksi haluatko kuvasta realistisen vai maalauksellisen? Kokeile!
- b. Kirjoita ensin runo valitsemastasi aiheesta. Kokeile seuraavaksi tehdä tekoälyn kanssa yhdessä runo Heureka näyttelystä löytyvän runokoneen kanssa.

Pohdittavaksi:

- Kummasta runosta pidät enemmän? Miksi?
- Millaista oli työskennellä yhdessä tekoälyn kanssa?
- Mihin muuhun voisi mielestäsi hyödyntää tekoälyavusteista kirjoittamista?

Täältä voit myös päästä kuuntelemaan Jukka Toivosen kehittämän runoja kirjoittavan tekoälysovelluksen P. O. Eticus-Apparatuksen runoutta:

<https://www.ruutu.fi/video/1163869>.

Löydät lisää Helsingin yliopiston Discovery-tutkimusryhmän taiteellisia tekoälysovelluksia täältä: <https://www2.helsinki.fi/en/researchgroups/computational-creativity-and-data-mining/art>



Tehtävä

Pystytkö tekoäly huijaamaan sinua?

Tekoälyohjelmista on tullut hyvin taitavia luomaan kuvia muun muassa ihmisistä ja paikoista, joita ei ole oikeasti olemassa. Näiden kuvien tekemiseen on käytetty samaa menetelmää, jolla Deepfake-huijaukset on myös tehty. GAN-menetelmässä toinen neuroverkko tuottaa kuvia ja toisen neuroverkon tehtävänä on tunnistaa oikeat kuvat keinotekoisista.

- a. Pystytkö sinä tunnistamaan oikeat ihmiskasvot tekoälyn luomista? Kokeile: <https://www.whichfaceisreal.com/index.php>

Pohdittavaksi:

- Mitä mahdollisia käyttötarkoituksia keksit tekoälyn tuottamille ihmisille?
- Mitä mahdollista haittaa tällaisesta teknologiasta on?

- b. Tässä tehtävässä tekoälymalli herättää tekemäsi piirroksen eloon.

Voit kokeilla toista tai vaikka molempia seuraavista vaihtoehdoista:

- Vaihtoehto 1: [pix2pix](#)-malli
 - Pääset testaamaan kissan, talon, käsilaukun tai kengän piirroksen muuttamista.
 - Ohjelma on helppo käyttää.
 - Sivusto on englanninkielinen, mutta tehtävien tekeminen ei juuri vaadi kielitaitoa.
- Vaihtoehto 2: [GauGAN2](#)-tekoälysovellus
 - Pääset luomaan maisemia piirroksesi pohjalta. Edistyneemmät voivat kokeilla myös valokuvien muokkaamista tekoälyn avulla.
 - Ohjelman käyttö vaatii hieman opettelua. Ohjelmasta tarjolla englanninkielisiä ohjevideoita.



Kuva: Ylhäällä ihmisen tekemä piirros ja alhaalla tekoälymallin muokkaama kuva

Heureka näyttelyssä voit myös kokeilla tekoölyavusteisesti piirtää kissan tai ihmisen.

Pohdittavaksi:

- Vastasiko tekoölyn luoma kuva omaa ajatustasi?
- Mihin tällaista kuvanmuokkausta voisi mielestäsi käyttää?

Lisämateriaalia:

- [AI with MIT App Inventor](#)
Mahdollisuus tehdä omia puhelimeen tai tablettiin ladattavia tekoölysovel-
luksia.
- [NVIDIA:n tekoölydemonstraatiota](#)
Täällä voit tutustua GauGAN2-lisäksi muihin NVIDIA:n tekoölydemonstraati-
oihin.
- [Evolution - Esimerkki evoluutioalgoritmista](#)
Evolution-ohjelmassa käyttäjä voi luoda oman ”otuksen” ja kokeilla opettaa
sitä juoksemaan, hyppäämään tai kiipeämään. Tämä havainnollistaa hyvin
evoluutioalgoritmien toimintaideaa.



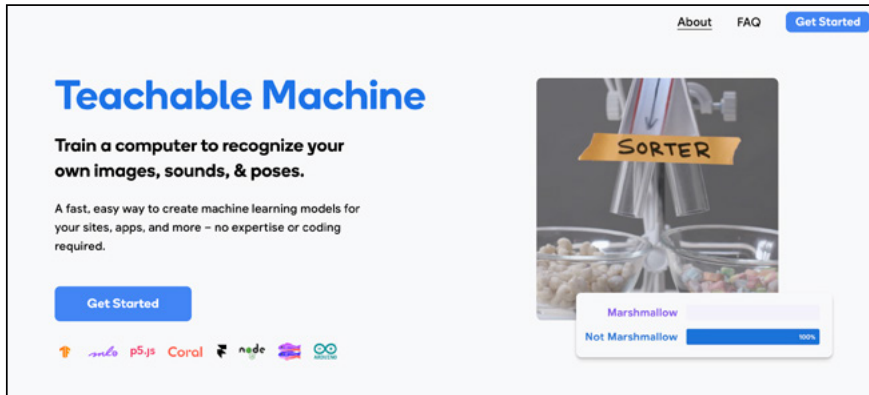
Oppimateriaalin ohjelmien käyttöohjeet



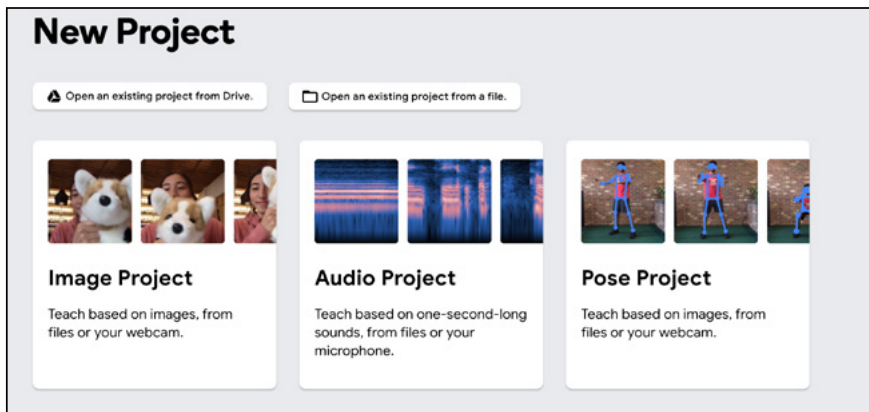
Kuva:
Google Teachable Machine

Ohjeet: Teachable Machine

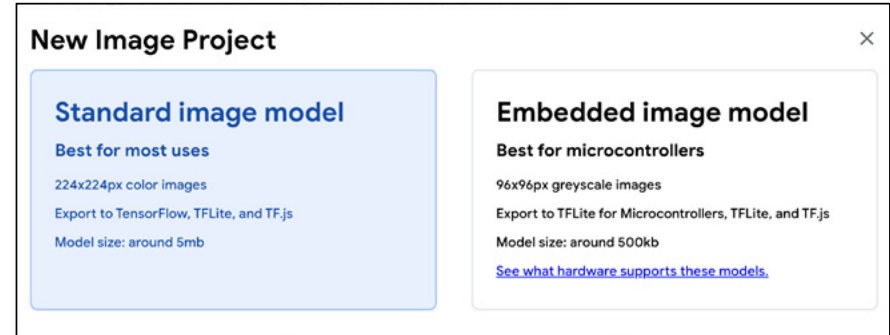
1. Luo uusi tekoälymalli Teachable Machine -työkalulla. Aloita mallin luominen painamalla **Get Started** -painiketta.



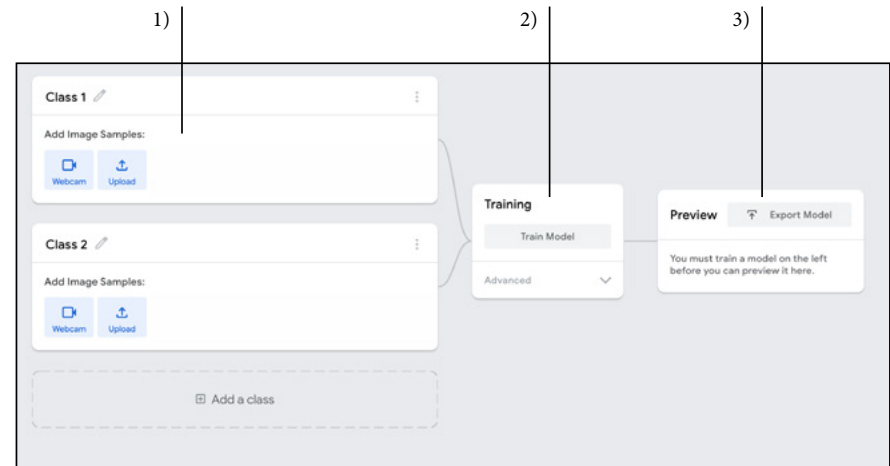
2. Voit valita, haluatko, että malli toimii kuvilla (**Image Project**), äänellä (**Audio Project**), tai asennon (**Pose Project**) kanssa. Tehdään tässä kohtaa kuvan tunnistus -projekti, joten valitaan vaihtoehto **Image Project**.



3. Valitse kuvamallin tyyppiä **Standard image model**.



4. Tekoälymallin luomisessa on kolme vaihetta: 1) opetusaineiston kerääminen, 2) tekoälymallin kouluttaminen ja 3) mallin lataaminen.

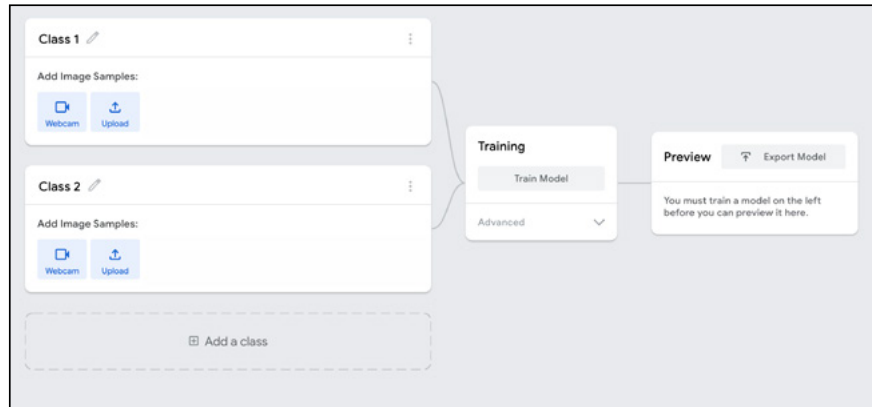


5. Tekoälymallissa kuvat tulee luokitella tiettyihin luokkiin. Esimerkissä banaanit luokiteltaisiin luokkaan 1 ja porkkanat luokkaan 2. Yleensä kannattaa tehdä myös yksi tyhjä luokka, johon lisätään taustakuvia (tai taustääntä).

Lisää näytteitä kaikille mallin luokille

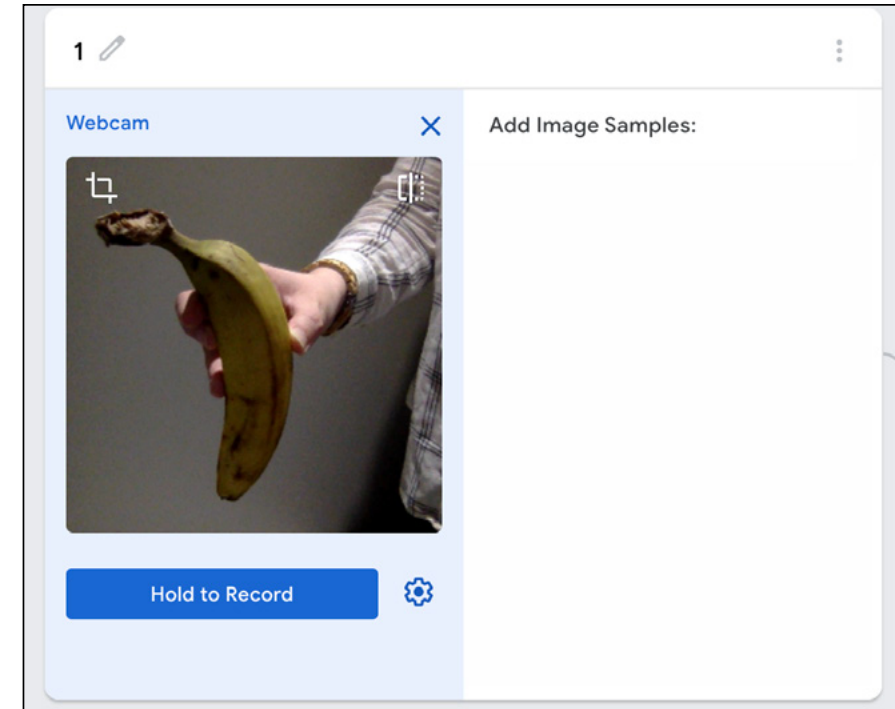
- Näytekuvienv kannattaa olla mahdollisimman monipuolisia, jotta tekoälymalli pystyy tunnistamaan sille myöhemmin näytettäviä kuvia.
- Ota näytekuvia **vähintään 40 kpl per luokka**.
- Nimeä luokat painamalla **Class 1, 2** jne. tekstin vieressä olevasta kynä-kuvakkeesta.
- Voit lisätä luokkia **Add a class** -painikkeesta.

Lisää luokkaan kuvia painamalla **Webcam**-painiketta.

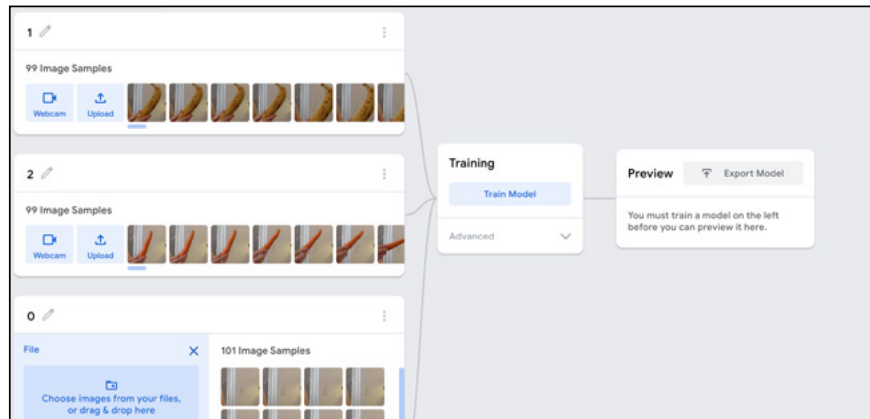


6. Paina **Hold to Record** -painiketta ottaaksesi kuvia lajiteltavasta esineestä.

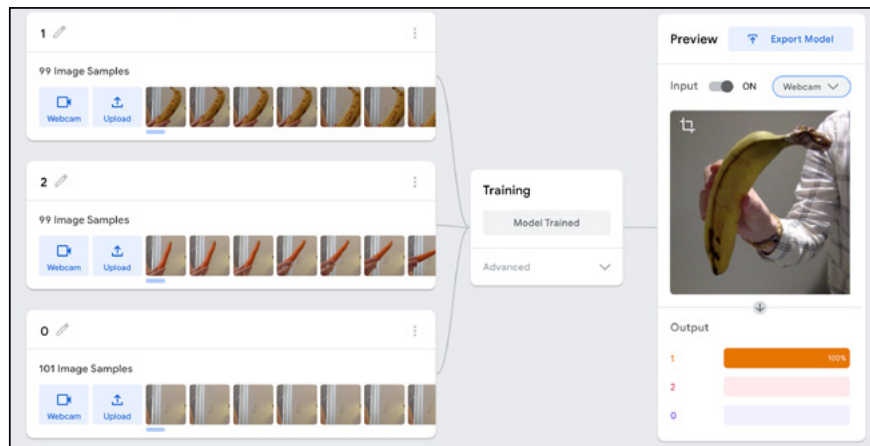
- Ota mahdollisimman erilaisia kuvia.
- Mitä kauemmin pidät **Hold to Record** -painiketta pohjassa, sitä useamman kuvan kamera ottaa peräkkäin.



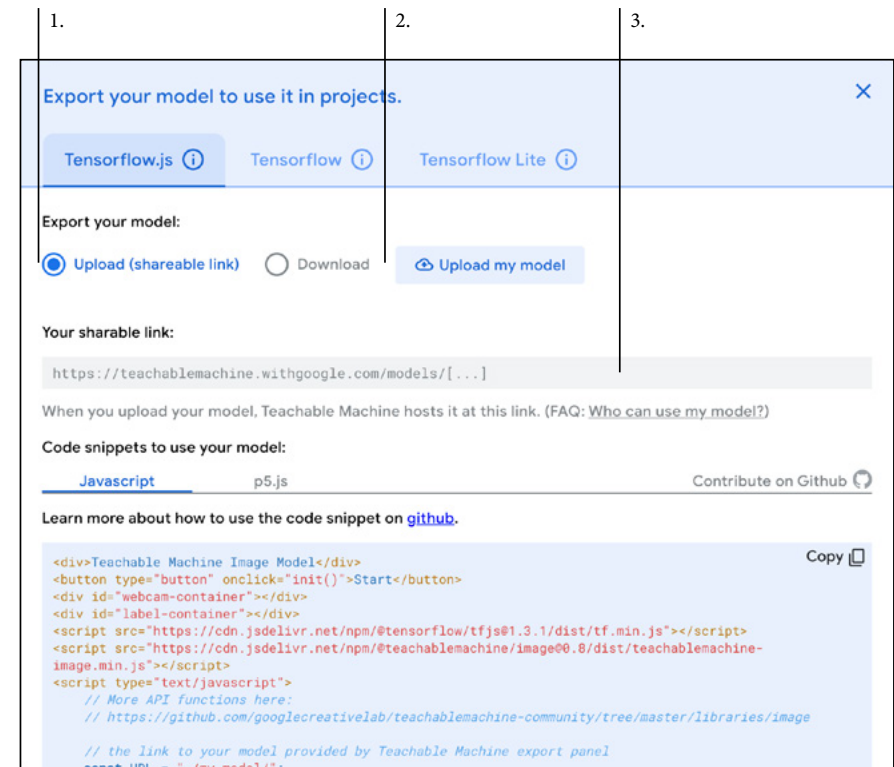
7. Kun olet valmis, paina **Train Model**-painiketta tekoälymallin kouluttamiseksi. Tässä saattaa kestää hetki.



8. Kun koulutusvaihe on valmis, paina **Export Model**-painiketta.



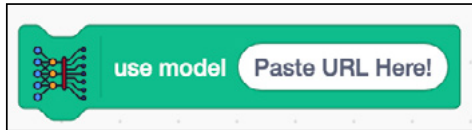
9.
 1. Valitse **Upload (shareable link)**
 2. Paina **Upload my model** -painiketta
 3. Kun lataus on valmis, kopioi linkki **Your shareable link** -kentästä.
 4. Voit tallettaa linkin itsellesi myöhempää käyttöä varten.



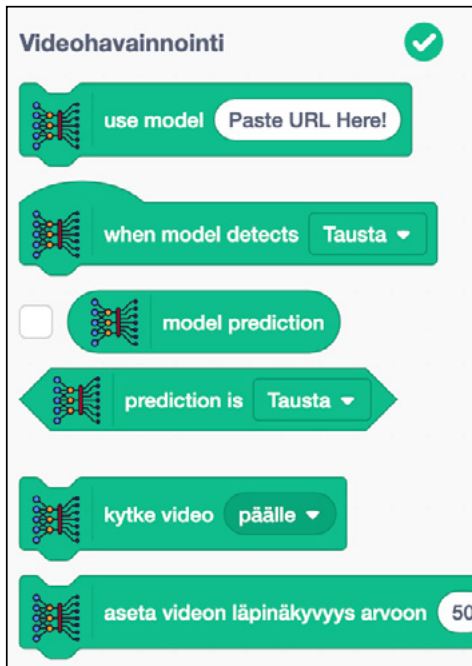
Ohjeet: AI Blocks

Liitä tekemäsi tekoälymalli omaan ohjelmointiprojektiisi [AI Blocks](#) -työkalulla.

1. Kopioi Teachable Machine -tekoälymallin linkki ja liitä se Videohavainnointi-välilehdeltä löytyvään **Use model** -palkkiin.

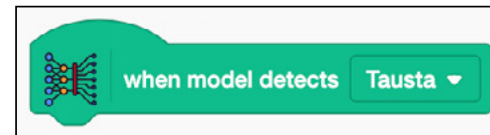


2. Klikkaa **Use model** -palkkia aktivoitaksesi mallisi.
3. Nyt Videohavainnointi-välilehden yläkulmassa pitäisi näkyä vihreä merkki oikeassa yläkulmassa.

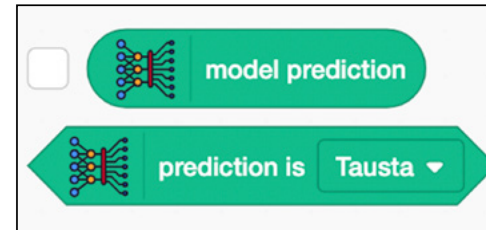


4. Voit käyttää tekoälymalliasi lisäämällä projektiisi:

- a. **When model detects** -palkin avulla voit koodata ohjelmasi tekemään jotakin, kun tekoälymallisi tunnistaa opettamasi luokan, kuten tietyn esineen tai eleen.



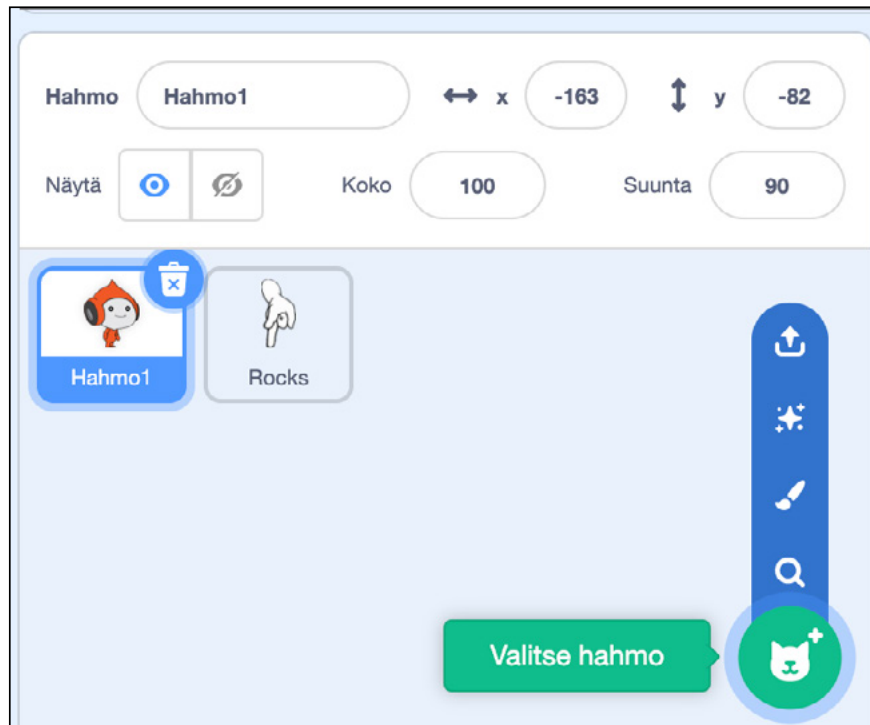
- b. **Model prediction** -palkin avulla ohjelma yrittää ennustaa, mihin tekoälymallillesi opettamista luokista näyttämäsi esine/ilme/liike/ääni kuuluu.



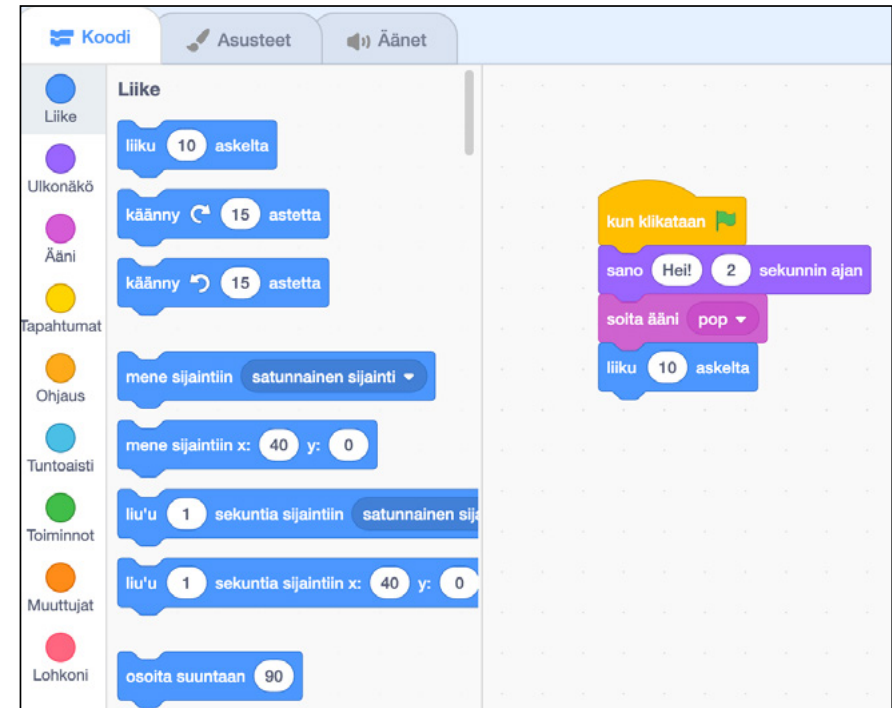
AI Blocks: yleisiä ohjeita

Hahmot

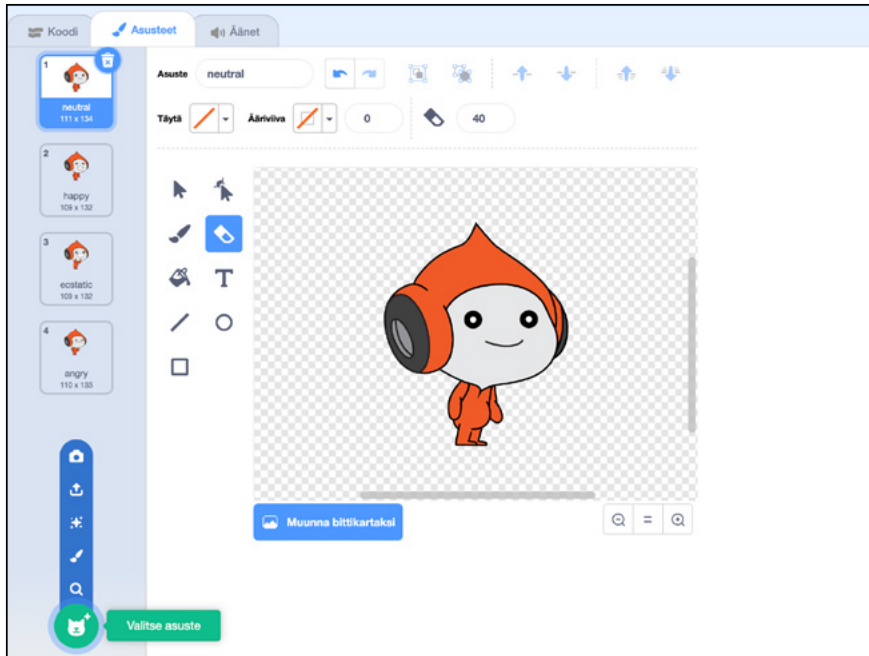
- Hahmoja voi lisätä tai vaihtaa painamalla oikealla alhaalla olevasta vihreästä kissa-painikkeesta.
- Hahmoja voi muokata erikseen. Vaihda hahmoa painamalla kerran sen painiketta.



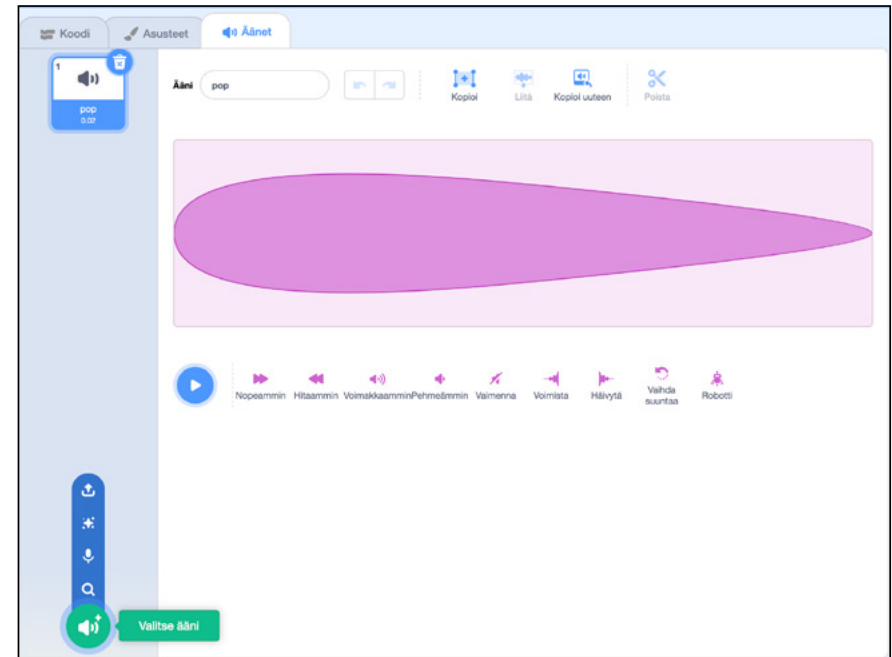
- Ylhäällä vasemmalla on kolme välilehteä: Koodi, Asusteet ja Äänet.
 - Koodi-välilehdellä voit koodata hahmon hyödyntämällä koodipalkeja.



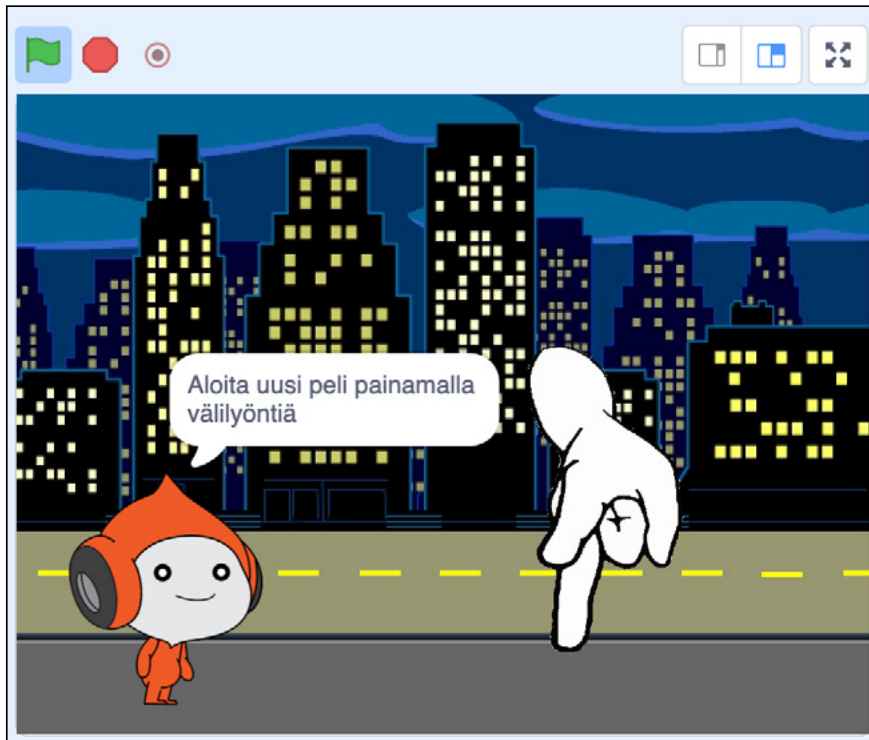
- Asusteet-välilehdellä voi muokata hahmoa.
 - Voit hakea lisää asusteita hakemistosta tai ladata niitä koneelta.
 - Voit myös muokata nykyistä hahmoa esimerkiksi muuttamalla väritystä tai piirtämällä.



- Äänet-välilehdellä voit lisätä ääniä valikosta tai äänittää niitä itse. Voit myös muokata ääniä.



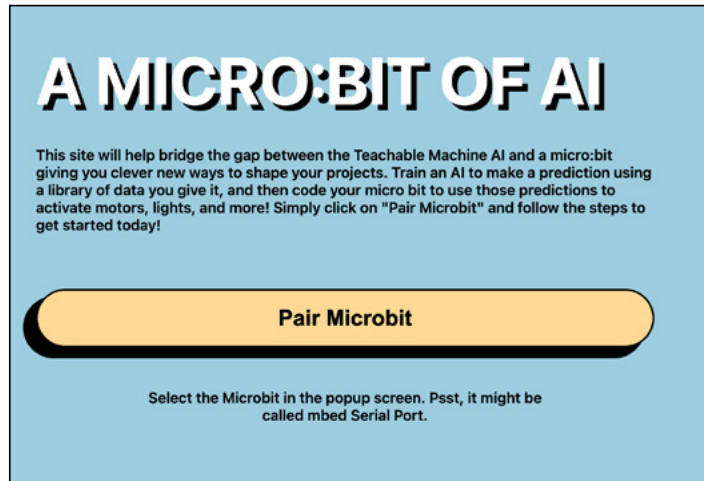
- Testaa ohjelmointia painamalla vihreää lippua näkymän vasemmassa ylälaudassa.



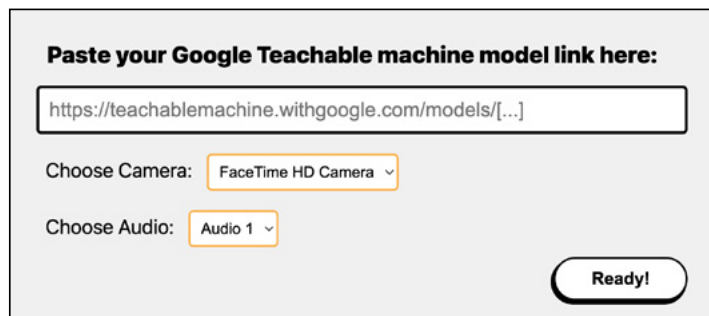
Ohjeet: Microbit of AI

Mene [Microbit of AI](#) -sivustolle.

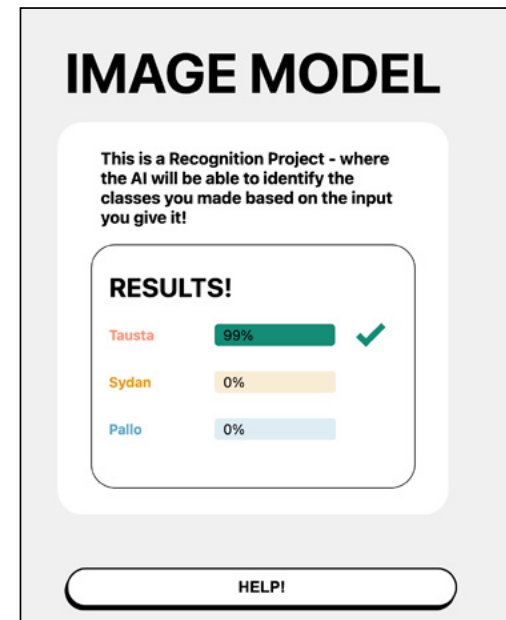
1. Paina **Pair Microbit** -painiketta. Valitse avautuvasta ikkunasta käyttämäsi Microbit ja paina **Muodosta yhteys** -painiketta.



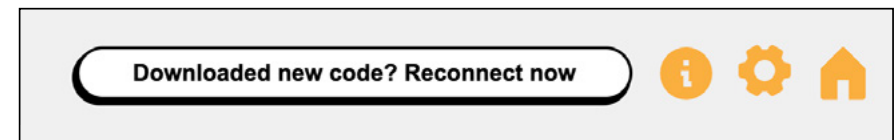
2. Kopioi Teachable Machine -tekoälymallin linkki ja liitä se sille varatulle riville. Tarkista, että kamera-asetukset ovat kunnossa. Kun olet valmis, paina **Ready!**



3. Nyt kaiken pitäisi olla valmista! **RESULTS!** -kohdassa voit tarkistaa, kuinka hyvin tekoälymallisi tunnistaa tekemäsi luokat. Tunnistetun luokan kohdalle tulee vihreä oikein merkki.



HUOM! Mikäli muokkaat ohjelmointiprojektiasi, niin muista päivittää kuvantunnistusohjelma painamalla:



Tekoäylajittelija Arduinolla (75–90 min)

Ohjelma on suunniteltu tukemaan Heureka Tekoäly – Me, Myself & AI -näyttelyn sisältöjä. Työpaja soveltuu peruskoulun yläluokkien ja toisen asteen teknologiakasvatukseen.

Ohjelman tavoite

Tekoälylajittelijan avulla opetellaan koneoppimista ja laskentaa konkreettisen rakenteluprojektin avulla. Googlen Teachable Machine -sivusto mahdollistaa ennalta opetettujen (“pre-trained models”) koneoppivien algoritmien hyödyntämisen omassa kokeilussa helpolla ja havainnollisella tavalla. Tavallisesti tällainen laskenta vaatii erittäin paljon laskentatehoa, mutta Teachable Machine -sivuston avulla datan luokittelu ja mallien opettaminen voidaan suorittaa helposti selaimessa.

Yläkoulun opetussuunnitelmassa (2014) tekoälyyn ja tarkemmin koneoppimiseen tutustuminen nivoutuu tieto- ja viestintäteknologisen osaamisen laaja-alaisiin tavoitteisiin (L5). Vaikkei tekoälyä mainita erikseen, ohjelmointia harjoitellaan eri oppiaineiden opinnoissa (mm. matematiikassa ja käsityössä). Oman koneoppivan algoritmin harjoittaminen ja sen soveltaminen eri ohjelmiin (P5 SerialControl ja p5.js - Web Editor sekä Arduino) tukevat suoraan *Käytännön taidot ja oma tuottaminen*, ja edelleen, *tiedonhallinta sekä tutkiva ja luova työskentely* -teemojen käytännön harjoittamista. Vuosiluokilla 7–9 matematiikan keskeisissä sisältöalueissa: “*Syvennetään algoritmista ajattelua. Ohjelmoidaan ja samalla harjoitellaan hyviä ohjelmointikäytäntöjä. Sovelletaan itse tehtyjä tai valmiita tietokoneohjelmia osana matematiikan opiskelua.*” (S1 Ajattelun taidot ja menetelmät).

Lukion opetussuunnitelmassa tekoälylajittelijan rakentaminen kytkeytyy laaja-alaisen osaamisen osa-alueista “monitieteisyys ja luova osaaminen” kokonaisuuteen. Lisäksi opetussuunnitelmassa on valinnaisten kurssien joukossa MAA11 Algoritmit ja lukuteoria, jonka keskeinen tavoite on tietää, mikä on algoritmi, sekä oppia tutkimaan, kuinka algoritmit toimivat. Edelleen valinnaisten kurssien puolelta löytyvät mm. ET6 Tulevaisuus, jossa kurssin keskeisinä sisältöinä ovat digitalisaatio, koneoppiminen, tekoäly ja robotiikka, sekä niiden vaikutus yhteiskuntaan ja kulttuuriin. Myös FI3 Yhteiskuntafilosofia -kurssilla keskeisenä sisältönä on ymmärtää teknologian ja tekoälyn vaikutusta yhteiskunnassa. Tarkoituksena on tutkia tekoälyä oman toiminnan kautta ja luoda sitä kautta uutta näkökulmaa tähän arkielämäämme vaikuttavaan ilmiöön.



Ohjeita materiaalin käyttöön

Työpajan kesto on 75–90 min.

Työpaja aloitetaan johdannolla erilaisista lajittelijoista, jotka toimivat ilman tekoälyä tai tekoälyä hyödyntäen. Tämän jälkeen toteutetaan oma kuvantunnistusta hyödyntävä tekoälylajittelija.

Tässä työpajassa toteutetaan tekoälylajittelija, joka tunnistaa ja lajittelee kuvia. Oma tekoälymalli on mahdollista harjoittaa tunnistamaan myös erilaisia ääniä. Äänilajittelijan tekemiseksi Teachable Machine -sovelluksessa tehdään Audio Project ja P5.js-editorissa käytetään tätä koodia.

Keskeiset käsitteet

tekoäly, koneoppiminen

Ohjelman ennakovalmistelut

On suositeltavaa, että koneelle asennettavat sovellukset ladataan ja asennetaan etukäteen ennen työpajan alkua.

Tarvikkeet

- Arduino Uno
- kytkentäjohtoja
- koekytkentälevy
- erivärisiä led-valoja
- vastuksia (330 Ω)
- USB-johto
- tietokone, jossa kamera tai liitettävä kamera
- lajiteltavia esineitä, esim. erivärisiä palikoita, hedelmiä

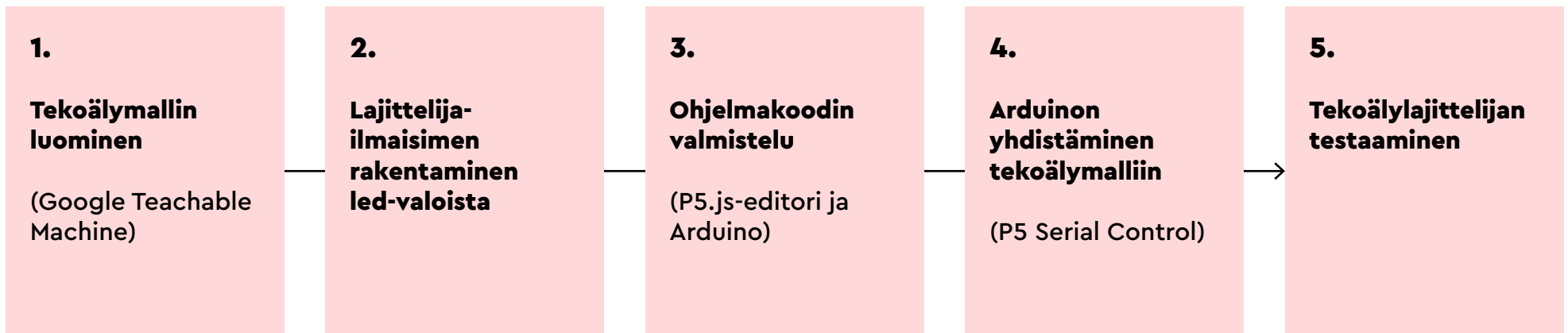
Käytettävät sovellukset ja ohjelmat

- [Google Teachable Machine](#) (selain)
- Arduino-ohjelmointiympäristö (selain tai asennettava), esim. [Codebender EDU Demo](#)
- P5.js -editori (selain)
- [P5 Serial Control](#) (asennettava)

Rakenne ja huomioita opettajalle

Tekoälylajittelijan luominen alkaa oman tekoälymallin opettamisella. Kun tekoälymalli on valmis, led-valoista rakennetaan lajittelijailmaisin. Tämän jälkeen valmistellaan ohjelmakoodi ja yhdistetään Arduino-kehitysalusta tekoälymalliin. Lopuksi testataan tekoälylajittelijan toimintaa.

Tekoäylajittelija Arduinolla -työpajan vaiheet



Huomioita työpajan eri vaiheista

1 Oman tekoälymallin luominen

Osiossa luodaan oma tekoälymalli [Googlen Teachable Machine](#) -työkalun avulla. Se on nettipohjainen työkalu, jolla voi tehdä erilaisia koneoppimismalleja helposti ja nopeasti. Nämä mallit voi sitten toteuttaa erilaisiin ohjelmiin ja alustoihin, kuten Arduinoon.

Ohjemateriaalissa on ohjeet kuvantunnistukseen, mutta projektin voi toteuttaa myös hyödyntäen äänen tai eleiden tunnistusta.

Ryhmästä riippuen tarvittaessa voi käydä läpi lyhyesti Teachable Machinen toiminnot. Halutessa ryhmille voi jakaa ohjeet paperisina. Seuraaviin seikkoihin kannattaa kiinnittää huomiota:

- Taustaluokan luominen parantaa oman mallin tarkkuutta
- Opetusaineiston luokan sisällä kannattaa olla monipuolisesti erilaisia sitä luokkaa edustavia kuvia. Esimerkiksi Banaaniluokkaan on hyvä ottaa kuvia banaanista eri etäisyyksiltä ja eri asennoista.

3 Ohjelmakoodin valmistelu

Voitte käyttää ohjelmointiin Codebender EDU Demo -sovellusta tai jotakin muuta Arduino-ohjelmointiympäristöä. Tässä ohjeessa käytetään Chrome-selaimessa käytettävää [Codebender EDU Demo](#) -sovellusta. Sovelluksen käyttämiseksi tarvitaan myös [Codebender App](#) -laajennus.

P5.js-editoriin ei tarvitse vaihtaa kuin kaksi asiaa: oman tekoälymallin linkki ja tietokoneen portti, johon Arduino on kytketty. Portin tietojen löytämiseen osa oppilaista saattaa tarvita apua. Mikäli portti ei löydy, niin kannattaa testata eri porttivalitukset läpi, kunnes koodi menee läpi.

4 Arduinon yhdistäminen tekoälymalliin

Oppilaille saattaa tulla tarve muokata koodia tässä vaiheessa. Kannattaa muistuttaa heitä sulkemaan portti P5 Serial Control -ohjelmassa ennen kuin he lähettävät muokatun koodin Arduinolle.

5 Tekoälylajittelijan testaaminen

Jos led-valot eivät syty, käynnistä p5.js-editori uudelleen.

Johdanto

Robottiikka-avusteista lajittelua voidaan hyödyntää erilaisissa tehtävissä. Esimerkiksi kirjastoissa on aineiston palautuksessa käytössä lajittelurobotteja, jotka osavat luokitella aineistoa teemoittain. Kirjaa, lehteä tai muuta aineistoa palautettaessa robotti lukee aineistoon kiinnitetyn RFID-tarran, jonka avulla se saa haettua kirjaston aineistotietokannasta tarkemmat tiedot aineiston sijainnista, esimerkiksi mihin kirjastoon ja mille osastolle aineisto kuuluu. Kun oikea sijainti on päätetty, aineisto ohjautuu liukuhihnaa pitkin oikeaan palautuslaatikkoon, josta kirjaston työntekijät vievät sen oikealle paikalle hyllyyn. Aineistojen tiedot on merkitty yksiselitteisesti, minkä vuoksi robotin on helppo lajitella ne oikeisiin laatikoihin.

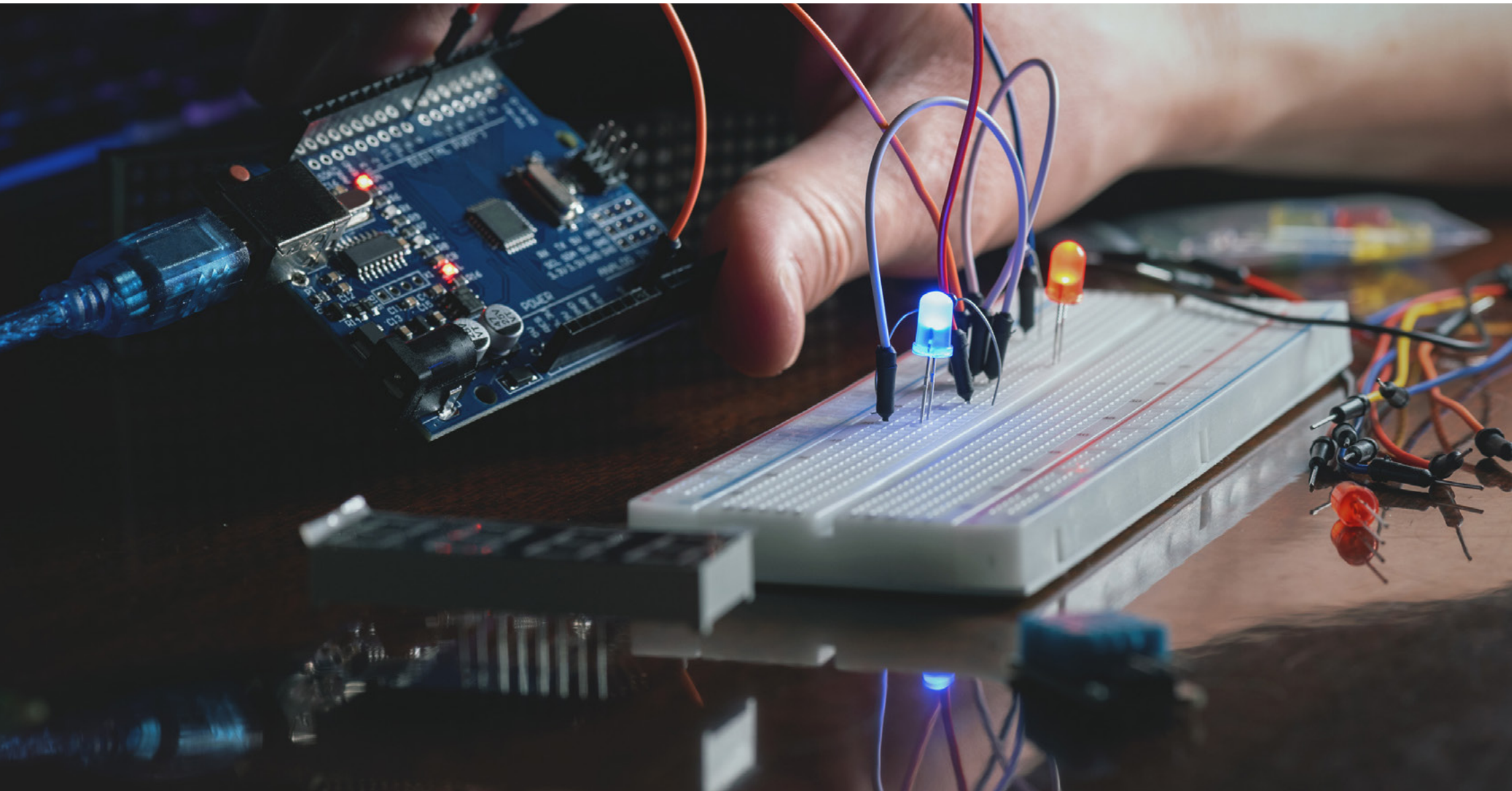
Helsingin keskuskirjasto Oodissa robotit toimittavat aineistoa jopa eri kerrosten välillä. Kun lajittelijarobotin lajittelema laatikko täyttyy, kuljetusrobotti kuljettaa laatikon hissiin ja sieltä edelleen kolmanteen kerrokseen. Jos hississä on ihminen, hissi antaa etusijan ihmiselle, mutta sen jälkeen hissi antaa etusijan kuljetusrobotille. Ja jos matkalla eteen tulee ihminen, robotti pysähtyy ja osaa laskea uuden reitin päämääräänsä. Kolmannessa kerroksessa kuljetusrobotti jättää laatikon kirjastolaisten hyllytettäväksi ja palaa hakemaan seuraavaa.

Jos lajiteltava esine ei ole helposti luokiteltavissa, voidaan lajittelussa hyödyntää tekoälyä. Esimerkiksi jätteenkäsittelylaitoksissa voidaan hyödyntää älykästä robotiikkaa rakennusjätteen tunnistamiseksi ja lajittelemiseksi. Robotin tehtävänä on erotella jätteen seasta omiin astioihin muun muassa metalli, puu ja kiviaines. Tässä kohtaa tehtävä on haastavampi kuin kirjojen lajittelun yhteydessä, sillä lajiteltutieto ei ole luettavissa materiaalista, vaan päätelmä täytyy tehdä sen ulkonäön perusteella. Ulkonäkökin saattaa erota, sillä jätejakeet voivat olla eri kokoisia, muotoisia tai värisiä. Robotti onkin etukäteen opetettu tunnistamaan jätteet valtavien kuvamäärän avulla ja opitun perusteella se lajittelee liukuhihnalta jätejakeet oikeisiin astioihin materiaalien uusiokäyttöä varten.

Jätteenkäsittelyssä lajittelurobotteja voidaan hyödyntää myös vaarallisen ongelmajätteen lajittelussa, jolloin ihmisen ei tarvitse niihin koskea. Tekoälyä hyödyntäviä lajittelurobotteja on etukäteen harjoitettu ongelmajättedatalla, jotta ne pystyvät toimimaan itseohjautuvasti muun muassa kuvantunnistusta hyödyntäen. Todellisuudessa robotti "näkee" lajiteltavat jakeet kameran kautta ja se vertaa kuvaa sille opetettuun aineistoon ja tekee päätelmän valinnasta, vaikka opetusaineistosta ei löydykään täysin samanlaista kuvaa.

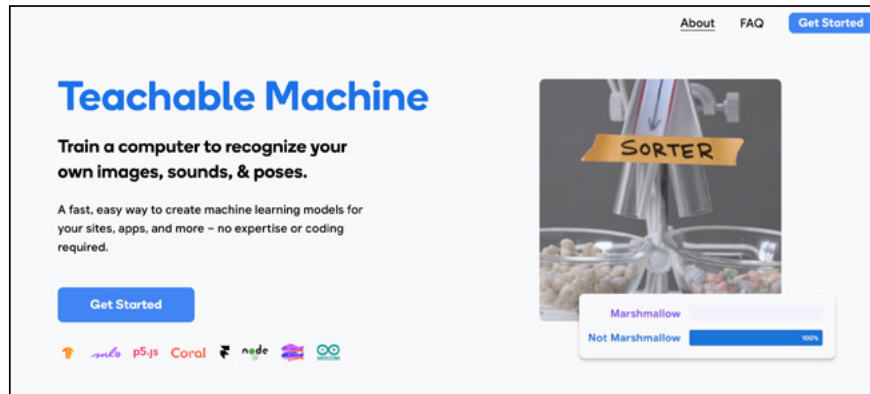
Tekoäly on osa datatiedettä, joka puolestaan sisältyy tietojenkäsittelytieteen alle. Kaplanin ja Haenleinin (2019) määritelmän mukaan se voidaan ymmärtää "järjestelmän kyvyksi tulkita ulkoisia tietoja oikein, oppia tällaisista tiedoista ja käyttää opittuja asioita tiettyjen tavoitteiden ja tehtävien saavuttamisessa joustavan sopeutumisen kautta."

Tekoälylajittelija Arduinolla -työohjeet

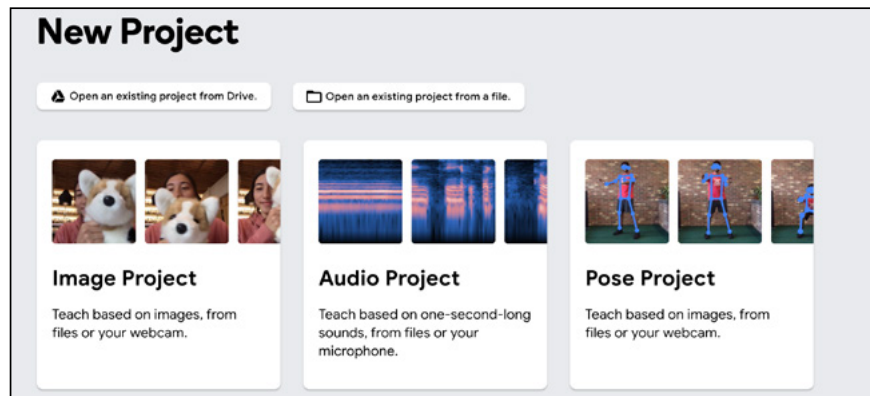


1 Oman tekoälymallin luominen

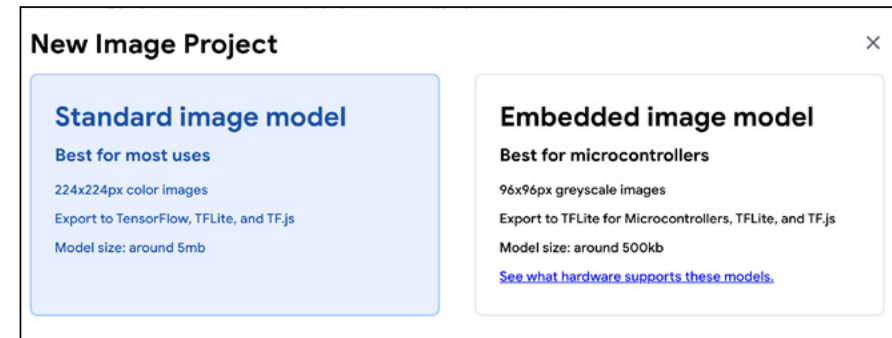
1. Luo uusi tekoälymalli Teachable Machine -työkalulla. Se on nettipohjainen työkalu, jolla voi tehdä erilaisia koneoppimismalleja helposti ja nopeasti. Aloita mallin luominen painamalla **Get Started** -painiketta.



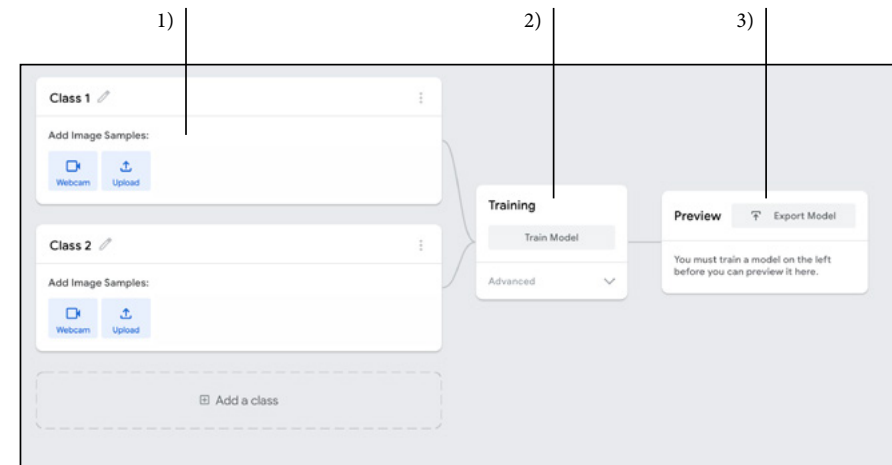
2. Voit valita, haluatko, että malli toimii kuvilla (**Image Project**), äänellä (**Audio Project**), tai asennon (**Pose Project**) kanssa. Tehdään tässä kohtaa kuvan tunnistus -projekti, joten valitaan vaihtoehto **Image Project**.



3. Valitse kuvamallin tyyppiä **Standard image model**.



4. Tekoälymallin luomisessa on kolme vaihetta: 1) opetusaineiston kerääminen, 2) tekoälymallin kouluttaminen ja 3) mallin lataaminen.

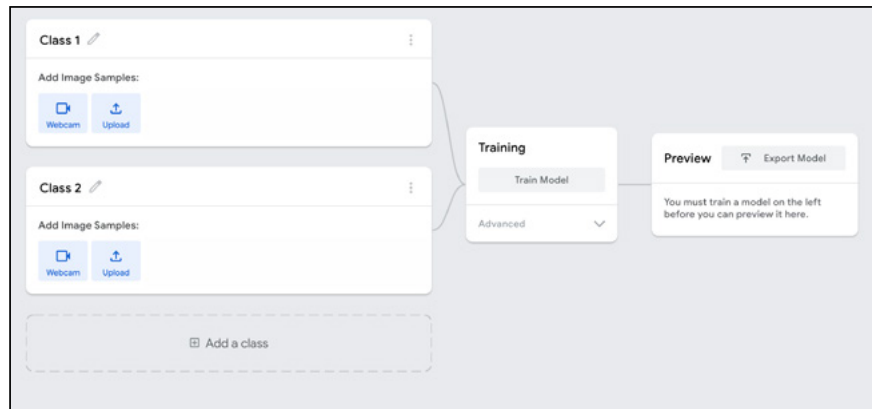


5. Tekoälymallissa kuvat tulee luokitella tiettyihin luokkiin. Esimerkissä banaanit luokiteltaisiin luokkaan 1 ja porkkanat luokkaan 2. Yleensä kannattaa tehdä myös yksi tyhjä luokka (luokka 0), johon lisätään taustakuvia.

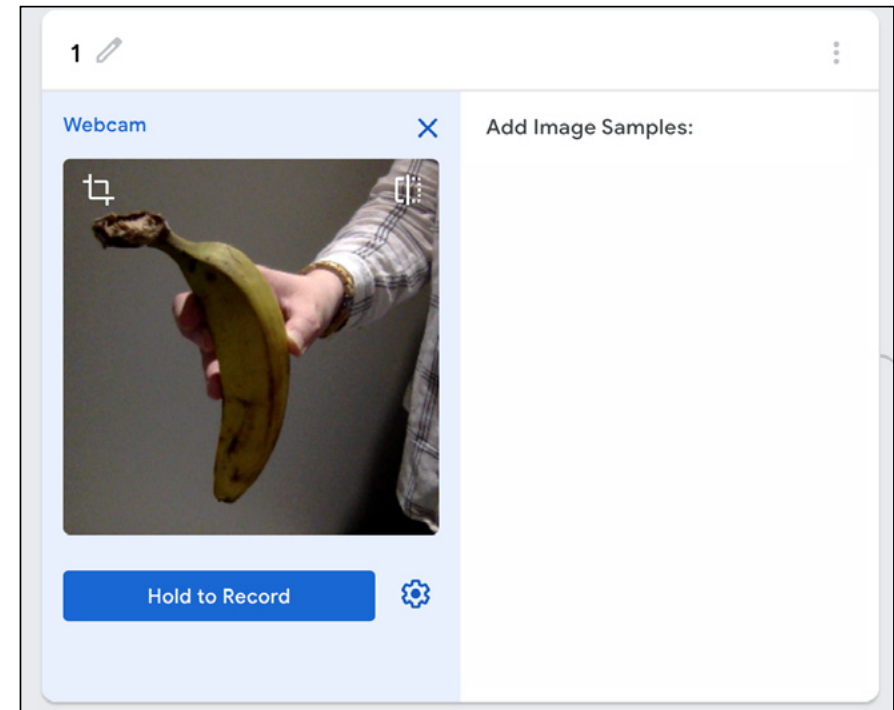
Lisää näytteitä kaikille mallin luokille seuraavasti:

- Näytekuvienv kannattaa olla mahdollisimman monipuolisia, jotta tekoälymalli pystyy tunnistamaan sille myöhemmin näytettäviä kuvia.
- Ota näytekuvia vähintään **40 kpl per luokka**.
- Nimeä luokat painamalla **Class 1, 2** jne. tekstin vieressä olevasta kynä-kuvakkeesta. Nimeä luokat pelkillä luvuilla, esimerkiksi "1", "2" ja "0" (tausta).
- Voit lisätä luokkia **Add a class** -painikkeesta.

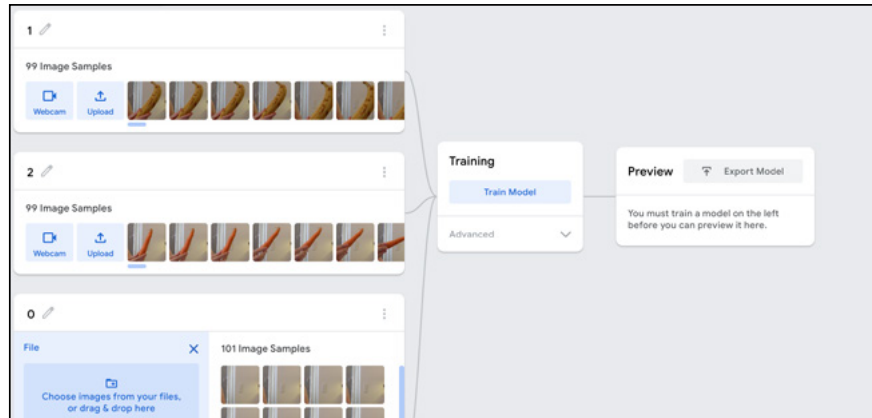
Lisää luokkaan kuvia painamalla **Webcam**-painiketta.



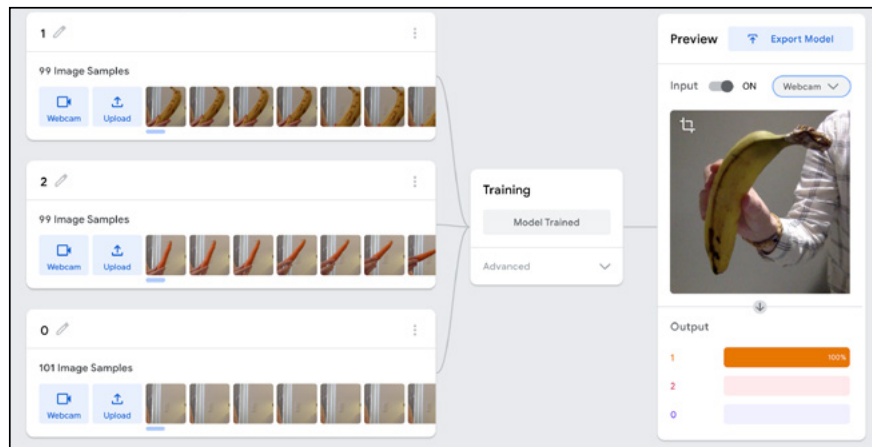
6. Paina **Hold to Record** -painiketta ottaaksesi kuvia lajiteltavasta esineestä
- Muista:
- Ota näytekuvat aina oikean luokan kohdalla. Esim. Kuvat banaanista luokan 1 kohdalla ja kuvat porkkanasta luokan 2 kohdalla.
 - Ota mahdollisimman erilaisia kuvia.
 - Mitä kauemmin pidät **Hold to Record** -painiketta pohjassa, sitä useamman kuvan kamera ottaa peräkkäin.



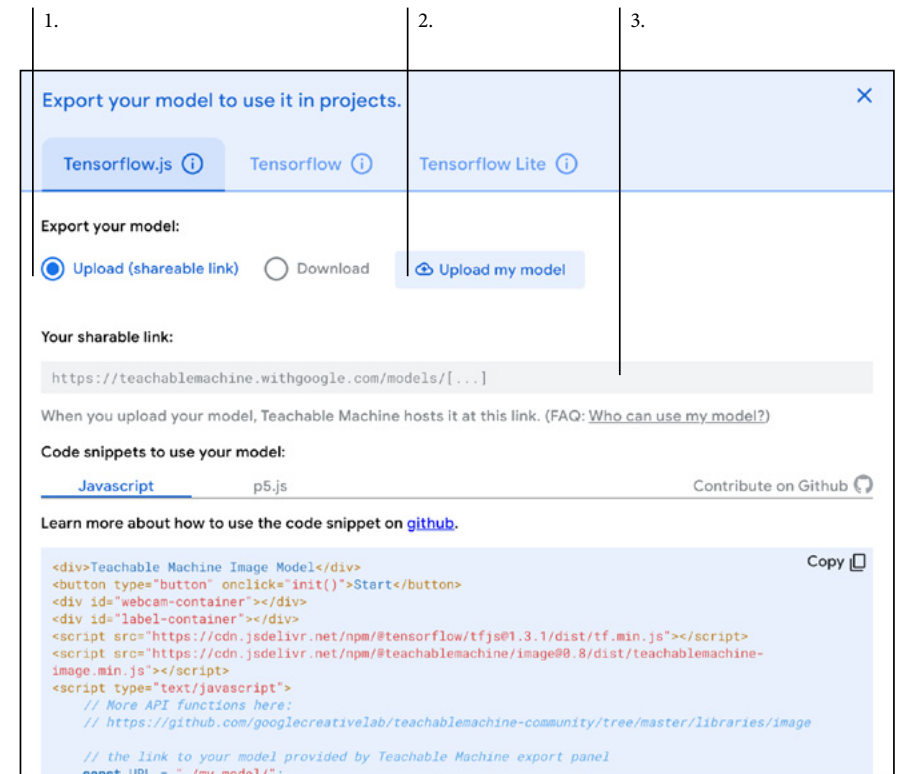
7. Kun olet valmis, paina **Train Model**-painiketta tekoälymallin kouluttamiseksi. Tässä saattaa kestää hetki.



8. Kun koulutusvaihe on valmis, paina **Export Model**-painiketta.



9.
 1. Valitse **Upload (shareable link)**
 2. Paina **Upload my model** -painiketta
 3. Kun lataus on valmis, kopioi linkki **Your shareable link** -kentästä.
 4. Voit tallettaa linkin itsellesi myöhempää käyttöä varten.

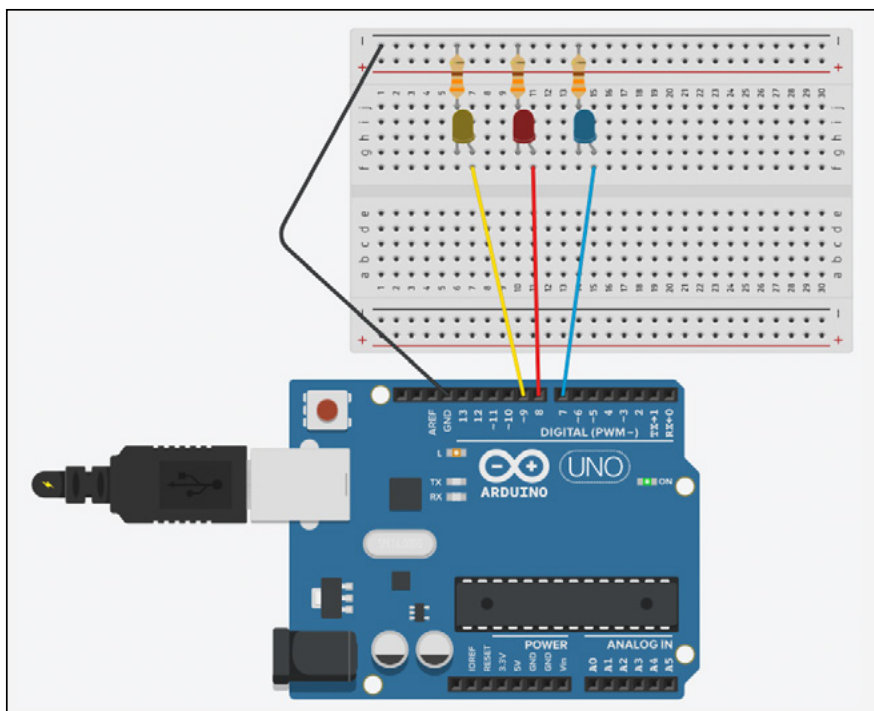


2 Lajittelijailmaisimen rakentaminen led-valoista

Seuraavaksi rakennetaan led-valoista virtapiiri, jonka kautta tekoälymalli ilmaisee, mihin luokkaan luokiteltava kuva kuuluu. Rakenna kuvien mukainen virtapiiri, jossa eriväriset led-valot tarkoittavat eri luokkia. Esimerkiksi keltainen valo tarkoittaa luokkaa 1, punainen led-valo luokkaa 2 ja sininen led-valo luokkaa 0 (tausta).

Tee seuraavat kytkennät:

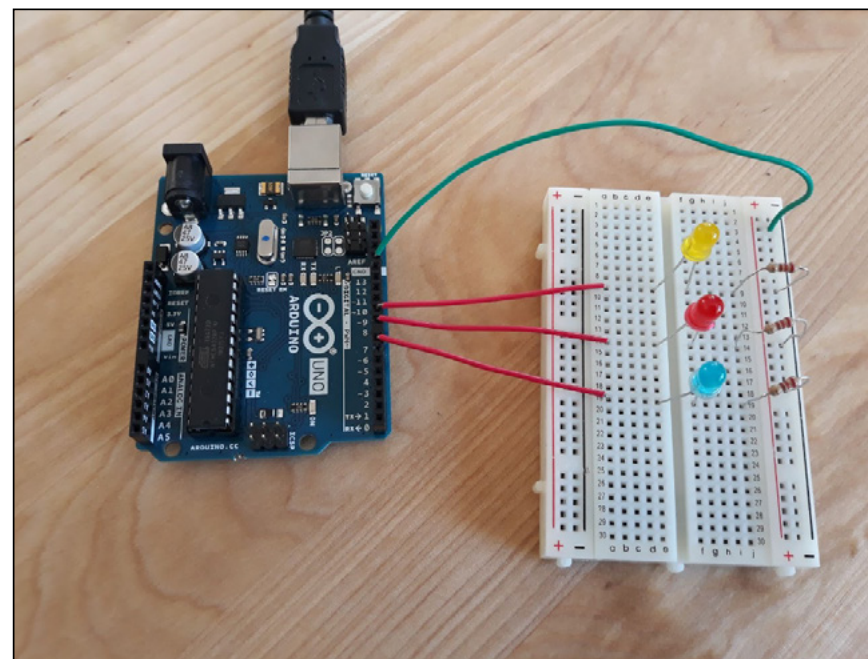
- luokkaa 1 edustava led-valo (tässä keltainen) kytketään digitaaliseen porttiin 9
- luokkaa 2 edustava led-valo (tässä punainen) kytketään digitaaliseen porttiin 8
- luokkaa 0 edustava led-valo (tässä sininen) kytketään digitaaliseen porttiin 7



Kuva: Kaaviokuva kytkennöistä. Kuvassa led-valon pidempi jalka (anodi) näkyy hieman taivutettuna.

Huomaa, että led-valoissa toinen jalka on pidempi kuin toinen. Pidempi jalka (anodi) kytketään plus-napaan, eli Arduinon digitaaliseen porttiin. Lyhyempi jalka (katodi) kytketään puolestaan miinus-napaan, eli maahan (GND-portti). Lyhyempi jalka kytketään ensin vastukseen ja sen jälkeen kytkentäjohton kautta maahan.

Lopuksi yhdistä Arduino-kehitysalusta koneeseesi USB-johdon avulla.

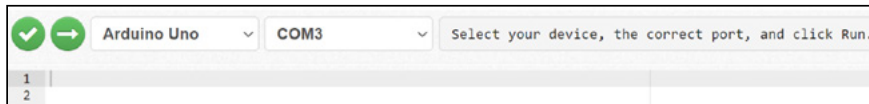


Kuva: Valokuva kytkennöistä.

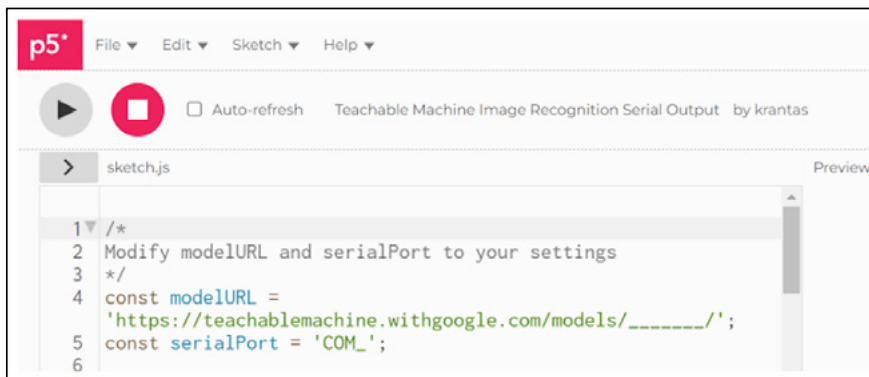
3 Ohjelmakoodin valmistelu

Käynnistä Arduinon ohjelmointiympäristö. Tässä ohjeessa käytetään Chrome-se-laimessa toimivaa Codebender EDU Demo -sovellusta, joka löytyy osoitteesta: https://edu.codebender.cc/class_demo. Sovelluksen käyttämiseksi tarvitaan myös Codebender App -laajennus, joka löytyy [täältä](#).

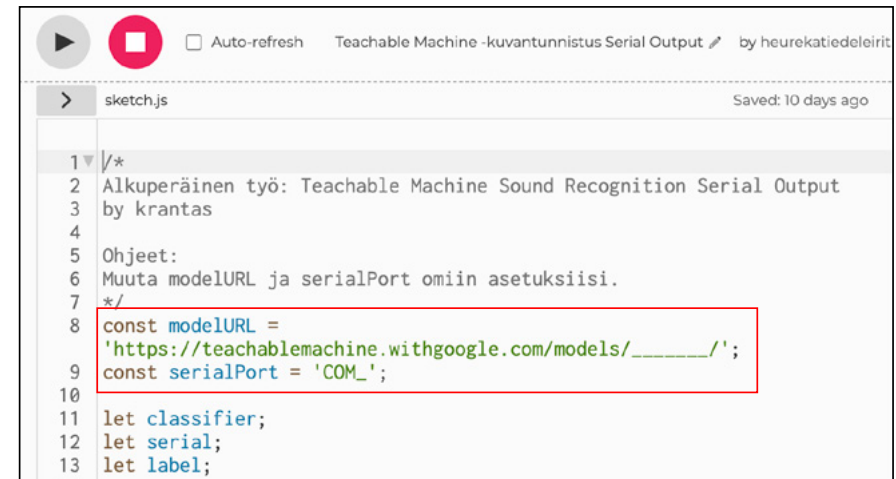
1. Tarkista Arduinon ohjelmointiympäristöstä tai koneesi laitehallinnasta, mi-hin tietokoneesi porttiin Arduino on kytketty. Ota portin nimi talteen.



2. Tekoälylajittelijan luomiseksi tarvitaan p5.js-editoria. [Käytä tätä valmista koodia](#), johon muokataan tekoälymallisi linkki sekä tietokoneesi portti, johon Arduino on kytketty.



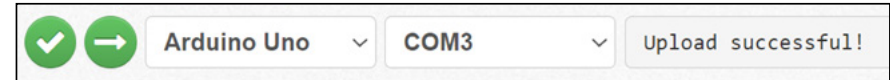
3. Vaihda koodissa kahdeksannen rivin **modelURL** oman tekoälymallisi linkiksi ja yhdeksannen rivin **serialPort** siihen tietokoneesi porttiin, johon Arduino on yhdistetty.



4. Palaa Arduino-ohjelmointiympäristöön ja kopioi alla oleva koodi ohjelmointiympäristöön.

```
char result='0';
void setup() {
  Serial.begin(9600);
  pinMode(9, OUTPUT); //luokan 1 portti
  pinMode(8, OUTPUT); //luokan 2 portti
  pinMode(7, OUTPUT); //luokan 0 portti
}
void loop() {
  while (Serial.available() > 0) {
    result = Serial.read(); // Lukee COM portin tulosta.
    switch (result) {
      case '1':
        digitalWrite(9, HIGH);
        digitalWrite(8, LOW);
        digitalWrite(7, LOW);
        break;
      case '2':
        digitalWrite(9, LOW);
        digitalWrite(8, HIGH);
        digitalWrite(7, LOW);
        break;
      default:
        digitalWrite(9, LOW);
        digitalWrite(8, LOW);
        digitalWrite(7, HIGH);
        break;
    }
  }
  delay (100);
}
```

5. Valitse ensimmäisestä pudotusvalikosta vaihtoehto **Arduino Uno** ja toisesta pudotusvalikosta portti, johon Arduino on kytketty (kuvassa COM3). Lähetä koodi Arduino-kehitysalustallesi painamalla **nuoli**-painikkeesta.

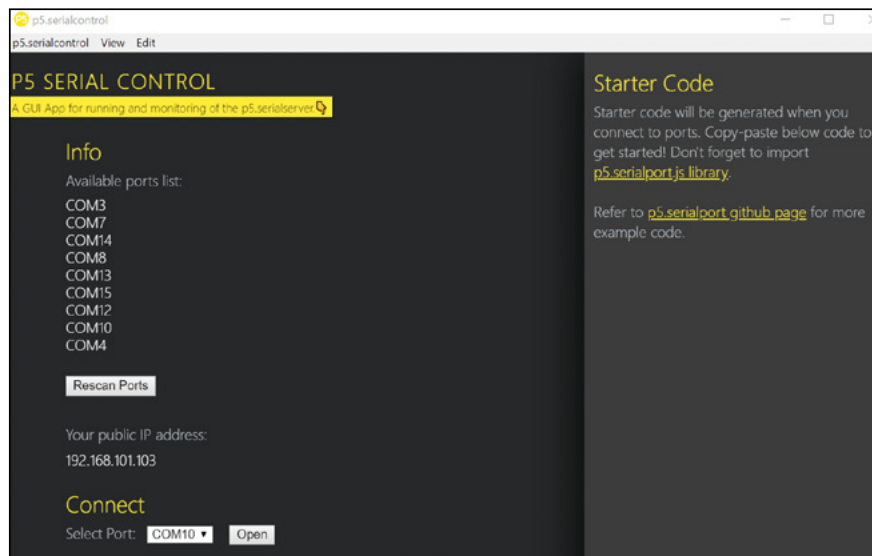


4 Arduinon yhdistäminen tekoälymalliin

Arduino yhdistetään Teachable Machinella tehtyyn malliin hyödyntämällä P5 Serial Control -ohjelmaa. Lataa ja asenna ohjelma koneellesi [täältä](#).

Avaa P5 Serial Control -ohjelma ja valitse Arduinon kytketty portti. Valitse portti kohdassa **Connect – Select Port** ja paina **Open**-painiketta.

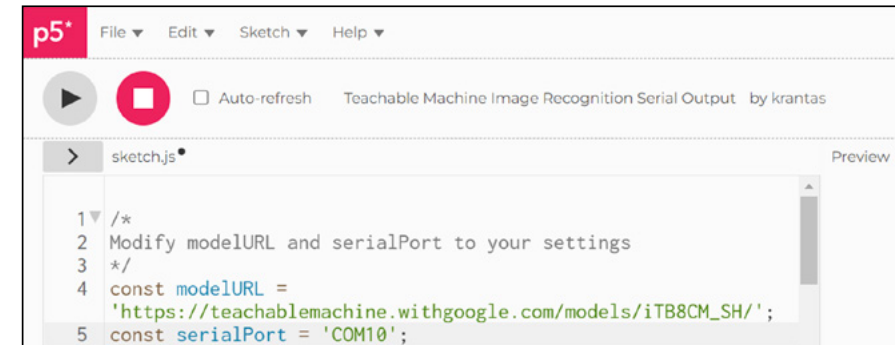
HUOM! Muista sulkea portti, jos yrität lähettää muokatun koodin Arduinolle.



5 Tekoälylajittelijan testaaminen

Käynnistä p5.js-editorin koodi. Voit ajaa koodin vasemmassa yläkulmassa olevasta **play**-painikkeesta. Salli ohjelman käyttää kameraasi.

HUOM! Tämän koodin tulee olla aina käynnissä, kun käytät tekoälylajittelijaa.



Tekoälylajittelijasi on nyt valmis testattavaksi! Vie lajiteltavia esineitä kameran eteen ja led-valot syttyvät sen mukaan, mihin luokkaan esine luokitellaan. Koodi lukee tekoälylajittelijan tulosta Serial.read() -komennolla ja sytyttää oikean värisen lampun riippuen siitä, mikä tulos on.

HUOM! Jos led-valot eivät lähde palamaan, kokeile käynnistää p5.js-editori uudelleen.

Pohdittavaksi:

- Mitä tapahtuu, jos koulutat tekoälymallin tunnistamaan ainoastaan yhden luokan?
- Miten voisit parantaa tekoälymallisi luotettavuutta?
- Miksi kannattaa käyttää eri esineitä testauksessa opetusaineistona käyttämiesi esineiden sijaan?

Työpajan pedagoginen toteutus on tehty yhteistyössä Vantaan Robotiikan oppimiskeskuksen kanssa.

Lähteet ja lisämateriaalia

- Ohje on mukailtu ohjeesta “Use Teachable Machine AI to Control Anything (CC BY-NC-SA)”. Englanninkielinen alkuperäisohje löytyy osoitteesta: <https://create.arduino.cc/projecthub/alankrantas/use-teachable-machine-ai-to-control-anything-2ad1ee>.
- Euroopan komission sivut. 5.3.2021. ZENROBOTICS: Applying artificial intelligence to waste sorting. https://ec.europa.eu/environment/ecoap/about-eco-innovation/good-practices/zenrobotics-applying-artificial-intelligence-waste-sorting_en
- Kaplan A. & Hainlein M. (2019). Siri, Siri, in my hand: Who’s the fairest in the land? On the interpretations, illustrations, and implications of artificial intelligence. <https://www.sciencedirect.com/science/article/abs/pii/S0007681318301393>
- Lukion opetussuunnitelma 2019: https://www.oph.fi/sites/default/files/documents/lukion_opetussuunnitelman_perusteet_2019.pdf
- Oodin sivut. 12.11.2018. Oodin robotit auttavat työn tekemisessä. <https://www.oodihelsinki.fi/oodin-robotit-auttavat-tyon-tekemisessa/>
- Perusopetuksen opetussuunnitelma 2014: https://www.oph.fi/sites/default/files/documents/perusopetuksen_opetussuunnitelman_perusteet_2014.pdf
- Uusiouutiset. 22.2.2022. Tekoälyrobotteja, oppivia koneita, pölytöntä ja hajutonta – Tältä näyttää maailman moderneimmassa kierrätyslaitoksessa Vantaalla. <https://www.uusiouutiset.fi/tekoalyrobotteja-oppivia-koneita-polytonta-ja-hajutonta-talta-nayttaa-maailman-moderneimmassa-kierratyslaitoksessa-vantaalla/>
- Wikipedia. [27.4.2022] Tekoäly (määritelmä): <https://fi.wikipedia.org/wiki/Teko%C3%A4ly>
- YLE. 24.7.2016. Robotti oppi lajittelemaan jätteet, mutta ihastui ensiksi kenkiin. <https://yle.fi/uutiset/3-9013193>

Tehtäviä Tekoälylajitteliija Arduinolla -työpajan tueksi



Tehtävä

Oman mekaanisen tekoälyn rakentaminen

Tarvikkeet (per pari):

- 10 kpl pelivälineitä (palikoita, kyniä, jäätelötikkuja tms.)
- 9 kpl kuppeja, jotka merkitty 2-10
- 26 kpl lappuja/palikoita/jäätelötikkuja tms.
- Tussi

Tavoite:

Tutustua tekoälyn opetusprosessiin

Tarkoituksena on tutustua tekoälyn oppimisprosessiin rakentamalla itse oma tekoäly. Tämän tekoälyn rakentamiseen ei tarvita lainkaan tietokonetta, vaan ainoastaan 9 kappaletta kuppeja sekä numeroidut laput.

Valmistelut:

- Merkitään kuppien kylkeen numerot 2-10
- Laitetaan jokaiseen kuppiin (numerot 3-10) kolme lappua, jotka on numeroitu 1, 2 ja 3.
- Laitetaan kuppiin numero 2 kaksi lappua, jotka on numeroitu 1 ja 2.

Taustaa

Mekaaninen tekoäly oppii tässä pelissä vahvistusoppimisen eli yrityksen ja erehdyksen kautta. Se sopii hyvin tilanteisiin, joissa tekoäly ei voi välittömästi tietää, oliko sen tekemä ratkaisu oikea vai väärä.

Pelissä kupissa oleva numero edustaa pelitilannetta eli pöydällä olevien pelivälineiden määrää. Kupissa olevat laput puolestaan edustavat tekoälyn mahdollisia siirtoja kussakin pelitilanteessa. Tekoäly valitsee seuraavan siirtonsa satunnaisesti kupissa olevista vaihtoehdoista.

Mikäli mekaaninen tekoäly häviää, siltä poistetaan tappioon johtanut siirto. Tällä tavalla siltä vähitellen poistetaan pelin kannalta huonoja ratkaisuvaihtoehtoja. Tekoälyn oppimisprosessia voisi nopeuttaa lisäämällä voittoon johtaneen pelin jälkeen voiton tuoneita lappuja jokaiseen kuppiin esimerkiksi 2. Näin todennäköisyys voitolle lisääntyisi jokaisen voiton jälkeen.

On syytä huomioida, että tällä järjestelyllä tekoäly opetetaan voittamaan, kun se itse aloittaa. Tällä tavalla opetettu tekoäly ei välttämättä voita, jos pelaaja aloittaa. Sen koulutuksesta puuttuvat tietyt pelitilanteet kokonaan. Tämän takia tekoälyä koulutettaessa ja käytettäessä on hyvä pitää mielessä, mihin kyseinen tekoälymalli on tarkoitettu.



Työohjeet

Tehtävänäsi on opettaa tekoäly pelaamaan peliä nimeltä **Nim**. Siitä on olemassa monta variaatiota, ja tämä on yksi sen yksinkertaisimmista muodoista. Tässä pelin säännöt:

- Aseta pöydälle 10 pelivälinettä
- Pelaajat nostavat vuorotellen 1–3 pelivälinettä pöydältä
- Pelin häviää se henkilö, joka nostaa pöydältä viimeisen pelimerkin.

Tee näin:

- Pelaa ensin parin kanssa muutama kierros. Kun pelin idea on tuttu, voit aloittaa tekoälyn kouluttamisen.
- Tarvitset 9 kuppia, joissa jokaisessa on valmiiksi numeroituja lappuja. Kupit on numeroitu 2–10. Kaikissa muissa kupeissa on kolme lappua, mutta kupissa numero 2 on ainoastaan kaksi lappua. **Osaatko sanoa miksi?**
- Tekoäly aloittaa pelin. Sen ensimmäinen siirto valitaan nostamalla yksi lappu kupista numero 10.
- Poista pöydältä kupista nostetun lapun osoittama lukumäärä pelivälineitä. Jätä lappu pöydälle sen kupin eteen, josta sen nostit.
- Seuraavaksi on sinun vuorosi poistaa pöydältä 1–3 pelivälinettä.
- Jatketaan vuorotellen, kunnes pelin voittaja on selvillä.

Kun peli päättyy:

- Jos tekoäly voitti, laita kaikki laput takaisin kuppeihin.
- Jos tekoäly hävisi, poista pelistä lappu, joka määrätti tekoälyn viimeisen siirron. Tämä oli huono siirto ja näin tekoäly oppii olla toistamatta sitä. Laita loput laput takaisin omiin kuppeihinsa.
- **HUOM: Jokaisessa kupissa on oltava koko ajan vähintään yksi lappu!** Jos tekoäly hävisi ja viimeisen siirron kupissa on ainoastaan yksi lappu, niin poista lappu sen sijaan sitä edellisen siirron kupista.

Jatka kouluttamista, kunnes olet tyytyväinen mekaanisen tekoälysi suoritukseen. Ideaalitulanteessa jokaisessa kupissa olisi ainoastaan yksi lappu.

Pohdittavaa:

- Miksi tekoälyä kouluttaessa et voi poistaa kupista viimeistä lappua?
- Onko tekoälyn voittoprosentti sama riippumatta siitä, aloittaako pelaaja vai tekoäly? Mikä voisi selittää tämän?
- Miten voisit tehostaa mekaanisen tekoälysi oppimista?
- Miten tekoälyn kouluttaminen pelaamaan Nim-peliä eroaa ohjelmoinnista?

Muokattu lähteestä: <https://www.i-am.ai/build-your-own-ai.html>



Tehtävä

Mitä YouTube oppii sinusta?

Tavoite:

- Oppia tunnistamaan erilaisia tekoälysovellutuksia arkielämän kontekstissa
- Tutustua erilaisiin algoritmeihin ja niiden taustalla oleviin vaikuttimiin

Välineet:

- Tietokone tai älylaite
- Taulukko

Opettajalle:

Oppilaat voivat toimia yksin tai pienissä ryhmissä. Oppilaiden tehtävänä on yrittää tunnistaa erilaisia algoritmeja, joita YouTube käyttää. Heidät voi johdatella teemaan esimerkiksi käymällä läpi ehdotettu haku (autofill) yhdessä. Tämän voi toteuttaa kirjoittamalla YouTuben hakukenttään ”kiss” ja katsoa, mitä vaihtoehtoja YouTube tarjoaa (kissavideo, kissing jne.). Tämän jälkeen voi tiedustella oppilailta, millaisesta tekoälysovelluksesta on kyse ja täyttää tiedot yhdessä lomakkeen ensimmäiselle riville.

Pohdintatehtävässä voi nostaa keskusteluun suosittelualgoritmien mukanaan tuomat haasteet, kuten flitterikuplat, kaikukammiot ja valeutiset.

Muokattu lähteestä:

Payne, B. H., yhteistyössä MIT Media Lab Personal Robots Group, johtajana C. Breazeal. (2019). An Ethics of Artificial Intelligence Curriculum for Middle School Students. (CC-BY-NC) <https://docs.google.com/document/d/1e9wx9oBg7CR0s5O7YnYHVmX7H7pnITfoDxNdrSGkp60/view#>

Lähteet ja lisätietoa

<https://yle.fi/aihe/artikkeli/2017/01/26/valheenpaljastaja-mita-valeutiset-ovat-ja-mita-ne-eivat-ole>

<https://yle.fi/aihe/artikkeli/2018/12/16/valheenpaljastaja-ihmiset-uskovat-somekupliin-mutta-vain-harva-elaa-oikeasti>

<https://policyreview.info/articles/analysis/should-we-worry-about-filter-bubbles>

Filterrikupla

Personoitu sisältö, joka on rajattu käyttäjän aikaisempien verkko-hakujen ja esimerkiksi sijaintitietojen pohjalta. Filterrikupla saattaa rajoittaa monipuolisen tiedon saantia.

Kaikukammio

Sosiaalisen median ilmiö, jossa samanmieliset henkilöt vahvistavat toistensa näkemyksiä.

Valeutinen

Harhauttamistarkoituksessa tehty video tai teksti, joka matkii journalistisen tuotoksen ulkoisia piirteitä. Tämä huijaus tai väärä tieto leviää etenkin sosiaalisessa mediassa.

Vastaukset

Nimi	Data	Mitä yrittää oppia?	Mitä Youtube optimoi?
Ehdotettu haku (autofill)	Aiempi hakuhistoria	Ennakoida, mitä haluat nähdä hyödyntämällä kaikkein yleisimpiä hakutuloksia	Katseltujen videoiden määrää
Suosituksiset	Käyttäjän ja muiden käyttäjien katseluhistoria	Mistä videoista saattaisit pitää aiemman katseluhistoriasi perusteella	Klikattujen videoiden määrää (lisäksi nähtyjen mainosten määrä)
Kommentit	Tykkäysten määrä, kuinka suosittuja kommenttien jättäjät ovat	Mitkä kommentit ovat tärkeimpiä	Mitkä kommentit pitävät käyttäjän pidempään sivustolla
Hakutulokset	Käyttäjän ja hänen kaltaisten käyttäjien aiemmat haut	Millaisia videoita käyttäjät haluavat nähdä tehdessään tietyn haun	Klikattujen videoiden määrää (lisäksi nähtyjen mainosten määrä)
Mainokset	Käyttäjän katseluhistoria	Mitkä tuotteet ja/tai palvelut saattaisivat kiinnostaa käyttäjää	Mainosten kohdentaminen
Automaattinen toisto	Käyttäjän ja muiden käyttäjien katseluhistoria	Minkä videon käyttäjä haluaa nähdä seuraavaksi	Katseltujen videoiden määrää, käyttäjien pitäminen sivustolla

Työohjeet

Tehtävänäsi on etsiä YouTube-sovelluksesta erilaisia tekoälysovelluksia.

Täytä löytämästäsi sovelluksesta tiedot seuraavan sivun taulukkoon.

Jos sinulla on vaikeuksia päästä alkuun, pohdi seuraavia asioita:

- Millaista tietoa tekoäly voisi YouTubessa kerätä?
- Mitä YouTube haluaa käyttäjän tekävän?
- Miten YouTube saisi käyttäjän toimimaan haluamallaan tavalla?

Pohdittavaksi

- Mitkä tekoälysovelluksista tekevät Youtubesta sinulle paremman? Mistä voisit luopua?
- Millaisia mahdollisia haasteita näet Youtuben käyttämissä tekoälysovelluksissa?
- Jos saisit suunnitella Youtube-sovelluksen aivan alusta, miltä se näyttäisi?

Vinkkejä lisämateriaaliksi tekoälystä ja supertietokoneista kiinnostuneille

Helsingin yliopisto & MinnaLearn. Elements of AI.

<https://course.elementsofai.com/fi/>

Helsingin yliopisto & MinnaLearn, Building AI.

<https://buildingai.elementsofai.com/>

Mariescu-Istodor. (2022). Self-driving car – Javascript course.

<https://youtube.com/playlist?list=PLB0Tybl0UNfYoJE7ZwsBQoDIG4YN9ptyY>

CSC – Tieteen tietotekniikan keskus & Kajaanin ammattikorkeakoulu. Johdatus supertietokoneisiin. <https://edukamu.fi/elements-of-supercomputing-fi>

Ihmisten tekoäly -ohjelmasarja, Tiedekulma

<https://www2.helsinki.fi/fi/tiedekulma/ohjelma/ihmisten-tekoaly>



Nimi	Data	Mitä yrittää oppia?	Mitä Youtube optimoi?

Lisätietoa

Koneoppiminen

Koneoppiminen on yksi suosituimmista tavoista tehdä tekoälysovelluksia. Se perustuu tietokoneen kykyyn oppia sille annetusta opetusaineistosta eli datasta. Data voi olla esimerkiksi kuvia, ääntä, tekstiä tai tykkäämisiä sosiaalisessa mediassa.

Data yksin ei riitä, vaan tarvitaan lisäksi tapa opettaa eli opetusalgoritmi. Näitä algoritmeja on tuhansia erilaisia ja niitä voi ikään kuin ajatella ohjelystoina, miten opetusaineiston kanssa tulee toimia. Opetustavan mukaan koneoppiminen voidaan jaotella kolmeen eri tyyppiin: ohjattu ja ohjaamaton koneoppiminen sekä vahvistusoppiminen.

Ohjatussa oppimisessa tekoälyjärjestelmää koulutetaan valmiiksi luokitellun opetusaineiston avulla. Siinä tavoitteena on, että tekoälymalli pystyy tekemään jaottelun samankaltaiselle aineistolle kuin mitä opetusaineisto on. **Ohjaamattomassa oppimisessa** tekoälymallilla ei ole olemassa valmista vastausta. Siinä tarkoituksena on tunnistaa aineistosta yhtäläisyyksiä ja ryhmitellä aineistoa sen pohjalta. **Vahvistusoppiminen** on yrityksen ja erehdyksen kautta oppimista. Sitä käytetään esimerkiksi itseohjautuvissa autoissa, joiden tulee pystyä toimimaan monimutkaisessa ympäristössä ilman välitöntä palautetta. Tämän oppimateriaalin kaksi työpajaa keskittyvät esittelemään ohjattua oppimista ja siitä tarkemmin luokittelua.

Toimivan mallin rakentamiseen tarvitaan monia toistoja ja laadukasta dataa. Työpajoissa käytettävä Teachable Machine -työkalu mahdollistaa nopean tekoälymallin koulutusprosessin. Teachable Machine käyttää siirto-oppimista, jossa tekoälymalli hyödyntää valmiiksi opetettua neuroverkkoa. Tämän vuoksi Teachable Machinella tehdyt tekoälymallit tunnistavat varsin luotettavasti esineitä jo suhteellisen pienillä kuva-aineistoilla.

Koneoppiminen ei ole kuitenkaan virheetöntä, vaan sekin on altis inhimillisille vaikutuksille. Ihminen valitsee opetusaineiston ja ohjaa, miten tekoälysovellus toimii sille annetussa tehtävässä. Ihminen saattaa esimerkiksi tiedostamattaan luoda

kasvojen tunnistussovelluksen, jonka opetusaineistona on käytetty pääasiassa valkoihoisia. Tämä opetusaineiston vinouma saattaa heikentää sovelluksen käytettävyyttä tummaihoisilla. Tämän vuoksi tekoälyn kehittäjien ja myös käyttäjien on hyvä olla tietoisia teknologian rajoitteista. Tekoäly on ainoastaan yhtä hyvä kuin sen kouluttamiseen käytetty aineisto.

Lähteet ja lisätietoja

Helsingin yliopisto & MinnaLearn. Elements of AI.
<https://course.elementsofai.com/fi/>

Helsingin yliopisto & MinnaLearn, Building AI.
<https://buildingai.elementsofai.com/>

Jyväskylän yliopisto. Tekoälyn perusteita ja sovelluksia.
<https://tim.jyu.fi/view/kurssit/tie/tiep1000/tekoalyn-sovellukset/kirja#DKUvbnUuGytQ>

Teachable Machine. FAQ. <https://teachablemachine.withgoogle.com/faq>

Tekoälyn etiikka

Tekoälyteknologia kehittyy vauhdilla, ja se on jo läsnä arjessamme. Se mahdollistaa meille asioita, joista aiemmin olemme voineet vain haaveilla, mutta samalla tarjoaa tilaisuuksia näiden mahdollisuuksien väärinkäyttöön. Tekoälyjärjestelmien kehittämisestä ja käyttämisestä vastaavat aina ihmiset ja tämän vuoksi ihmisten on tärkeä olla tietoisia, millaisia eettisiä kysymyksiä tekoälyteknologioihin liittyy.

Tekoälyn etiikassa on kyse arvoista, joiden pohjalta tekoälyteknologiaa kehitetään ja käytetään. Tekoälyjärjestelmiä ei kehitetä muusta maailmasta irrallaan vaan niihin vaikuttavat monet tekijät, kuten kuka ne on suunnitellut, miksi ne on kehitetty, mihin niitä käytetään, millaisilla algoritmeilla ne toimivat ja millaisella opetusaineistolla ne on koulutettu.

Monet julkiset ja yksityiset tahot ovat luoneet omia eettisiä periaateohjelmia ohjaamaan tekoälyn kehittämistä ja käyttöä. Ohjelmien välillä on jonkin verran eroavaisuuksia, mutta pari vuotta sitten tehdyn tutkimuksen (Jobin ym., 2019) mukaan seuraavilla periaatteilla näyttäisi olevan laaja kannattajakunta:

- oikeudenmukaisuus
- vastuullisuus
- läpinäkyvyys
- vahinkojen välttäminen
- ihmisoikeuksien kunnioittaminen (mm. oikeus yksityisyyteen ja valinnanvapauteen).

Tekoälyn eettisyys ei ole vain yhden tahon vastuulla, vaan se on tärkeä huomioida kehitysvaiheesta lähtien aina loppukäyttäjään saakka. Viranomaisilla on toki tärkeä rooli teknologian säännöstelyn näkökulmasta ja Euroopan unioni on tätä kirjoitettaessa (vuonna 2022) laatimassa omia ohjeistuksiaan tekoälyteknologian sääntelyyn. Teknologia itsessään on harvoin sypää, vaan eettiset haasteet syntyvät ihmisten toimien kautta. Monet haasteet kulminoituvat kysymyksiin, mihin ja miten kehitettyä teknologiaa käytetään. Esimerkiksi GPT-2- tekoälymallin kehittänyt Open.AI totesi vuonna 2019, ettei maailma ollut vielä valmis heidän kehittyneelle kielimallilleen. Tämän vuoksi he julkaisivat sen ensin neljäsosan kokoisena mi-

niversiona alkuperäisestä. He pelkäsivät, että sitä voitaisiin käyttää vihamielisessä tarkoituksessa esimerkiksi tuottamaan valeuutisia tai tietojenkalasteluhuijauksia.

Opetusaineisto vaikuttaa merkittävästi kehitetyn tekoälyn eettisyyteen. Tekoälyn kouluttamista varten data pitää saada käyttövalmiiksi eli kerätä, käsitellä ja tallentaa. Jos näitä vaiheita ei tehdä eettisesti kestävästi, ei datan pohjalta koulutettu tekoälykään voi olla eettinen. Datassa saattaa olla vinoumia, jos sitä ei ole kerätty kattamaan koko haluttua kohderyhmää. Tämä saattaa johtaa pahimmillaan algoritmiseen syrjintään. Esimerkiksi työhakemuksia seulovan algoritmin kouluttamiseen käytettävässä datassa on syytä kiinnittää huomiota siihen, miten työpaikkavalinnat on aiemmin tehty. Algoritmi ei itsessään syrji, mutta jos se altistuu esimerkiksi enemmän miehiä sisältävälle aineistolle, se saattaa oppia suosimaan miehiä valintapäätöksissään. Algoritminen päätöksenteko on tällä hetkellä haastavaa myös siitä syystä, ettei tuloksesta välttämättä voi päätellä, miksi algoritmi tuotti juuri tämän tuloksen.

Turvallisuusnäkökohtiin liittyy esimerkiksi se, mihin kaikkeen meistä kerättyä tietoa käytetään ja miten sitä yhdistellään. Tämän vuoksi kaikki verkkosivut kysyvät nykyään käyttäjiltä lupaa tallentaa heidän tietonsa, ja käyttäjillä on myös oikeus saada halutessaan tietoonsa kaikki heistä kerätty tieto. Tekoälyyn liittyy monia vastuukysymyksiä, kuten kuka on vastuussa, jos tekoälyn käyttämä algoritmi tekee päätöksen väärin perustein. Tekoälyä saatetaan myös ilman säätelyä käyttää vihamielisiin tarkoituksiin esimerkiksi hyödyntämällä Deepfakea ja saada henkilö sanomaan jotain videon tekijän tarkoitukseen sopivaa.

Lähteet ja lisämateriaalia:

Jobin, A., M. Ienca & E. Vayena (2019). The global landscape of AI ethics guidelines. *Nature Machine Intelligence* (ISSN 2522-5839), vol. 1, pp. 389-399. Saatavilla:

<http://ecocritique.free.fr/jobin2019.pdf>

<https://ethics-of-ai.mooc.fi/fi/>

<https://openai.com/blog/better-language-models/>

<https://www.aimyths.org/ai-can-be-objective-or-unbiased>

<https://www.sitra.fi/blogit/eettinen-tekoaly-tarvitsee-eettista-dataa/>

https://aisociety.fi/sites/aisociety.fi/files/opas_tekoalyn_etiikkaan_v1.pdf

<https://www.aimyths.org/ethics-guidelines-will-save-us>

Supertietokoneet tekoälyn kehityksen taustalla

Tekoälyn jatkuva kehittyminen vaatii paljon laskentatehoa ja siinä otettavat edistysaskeleet ovat osittain riippuvaisia tietokoneiden kehityksestä. Tekoälyjärjestelmiä kehitettäessä yksittäinen perustietokone ei useimmiten riitä vaan käyttöön on otettava supertietokone.

Supertietokone koostuu samoista perusosista kuin kotitietokonekin. Itse asiaan uusin teknologia löytyy suuren kysynnän vuoksi supertietokoneiden sijaan kuluttajille suunnatuista tietokoneista. Suuri ero laskentatehossa johtuu laitteiden mittakaavasta. Supertietokoneen muisti voi olla miljoonia gigatavuja, kun kannettavassa tietokoneessa 32 gigatavua on jo iso muistikapasiteetti.

Mittakaavasta kertoo jotain myös se, että tavallisessa kotitietokoneen keskusyksikössä (CPU) on tyypillisesti 4–8 ydintä, kun taas supertietokoneessa niitä voi olla jopa satoja tuhansia. Tietokoneen keskusyksikkö suorittaa tietokoneohjelman ohjeita ja sen vuoksi siihen välillä viitataan myös suoritin-nimellä. Huippupelittietokoneissa on tehokkaat grafiikkasuorittimet, mutta supertietokoneessa grafiikkasuorittimia on jopa tuhansia. Supertietokoneen laskentateho perustuu kaikkien suoritusydinten ja grafiikkasuorittimien samanaikaiseen toimintaan.

Supertietokoneet voivat merkittävästi lyhentää tekoälysovellusten koulutusaikaa. Esimerkiksi eturauhassyövän kudoksenäytteestä tunnistavan tekoälyjärjestelmän koulutukseen tarvittiin kuvadataa noin 30 teratavua. Tavallisen tietokoneen laskentateholla koulutus olisi kestänyt vähintään kuukausia. Supertietokone hoiti saman tehtävän 2–3 päivässä.

Tekoälyn kielenkäsittelykyky on ottanut suuria harppauksia viime aikoina neuroverkkojen ja syväoppimisen avulla. Nykyiset neuroverkot voivat ymmärtää sanastollisen kontekstin ja myös muistaa aiemmin syötettyjä sisältöjä. Edistyneiden kielimallien kouluttaminen vaatii todella suuria määriä tekstiä, jonka käsittely ei ole helppoa ilman supertietokoneita.

Supertietokoneet eivät ole kuitenkaan ainoa vaihtoehto syväoppimistekoälyn kouluttamiseen. Grafiikkasuorittimet mahdollistavat neuroverkkojen rinnakkais-

laskennan. Rinnakkaislaskennan avulla pystytään toteuttamaan hyvin suuriakin syväoppimisalgoritmeja ilman supertietokoneita. Se on menetelmä, jossa ongelma jaetaan pienempiin aliohjelmiin. Aliongelmien laskenta suoritetaan samanaikaisesti ja niiden tulokset yhdistetään ja saadaan ratkaisu. Tämä lyhentää huomattavasti laskenta-aikaa. Supertietokoneissa rinnakkaislaskenta vähentää niiden kuluttamaa virtaa.

Tiesitkö, että Euroopan tehokkain tietokone löytyy Kajaanista?

LUMI-supertietokone sijoittuu maailman nopeimpien supertietokoneiden listan sijalle kolme.

Lähde ja lisämateriaalia:

<https://www.heureka.fi/mika-on-supertietokone/>

CSC – Tieteen tietotekniikan keskus & Kajaanin ammattikorkeakoulu. Johdatus supertietokoneisiin. <https://edukamu.fi/elements-of-supercomputing-fi>

Helsingin yliopisto & MinnaLearn. Tekoälyn perusteet: Neuroverkkojen periaatteet. <https://course.elementsofai.com/fi/5/1>

Tekoäly - Me, Myself & AI on tehty yhteistyössä tekoälyä
tutkivien ja kehittävien yhteistyökumppaneiden kanssa.
Lämmin kiitos!



efima

FCAI

futurice

GOFORE



HELSINGIN YLIOPISTO



KORKEASTI KOULUTETTUJEN KASSA

 **Microsoft**

NOKIA

 **SOLITA**

STEK

TEK