Akseli Kankaansivu

Tekoälyn vaarat: koneoppimismalleihin perustuvien järjestelmien haavoittuvuudet ja niiltä puolustautuminen

Tietotekniikan kandidaatintutkielma

26. huhtikuuta 2023

Jyväskylän yliopisto

Informaatioteknologian tiedekunta

Tekijä: Akseli Kankaansivu

Yhteystiedot: kankaazt@student.jyu.fi

Ohjaaja: Timo Tiihonen

Työn nimi: Tekoälyn vaarat: koneoppimismalleihin perustuvien järjestelmien haavoittuvuudet ja niiltä puolustautuminen

Title in English: The dangers of A.I; how to protect a machine learning-system from its vulnerabilities

Työ: Kandidaatintutkielma

Opintosuunta: Tietotekniikka

Sivumäärä: 31+0

Tiivistelmä: Tekoälyn laajamittainen käyttö nykypäivänä lisää innovaatioita ja tehokkuutta, mutta tuo samalla esiin uusia haavoittuvuuksia, joita voidaan hyödyntää erilaisissa hyökkäyksissä. Tämä tutkielma analysoi yleisimpiä tekoälyyn liittyviä kyberturvariskejä ja havaitsee, että tekoälyn oppimisprosessi on sen haavoittuvin osa-alue. Tutkielmassa ehdotetaan ulkoisien tekoälyjärjestelmien käyttöä näiden heikkouksien torjumiseksi. Tutkielma on toteutettu kirjallisuuskatsauksena.

Avainsanat: Tekoäly, Haavoittuvuus, Data poisoning, Kyberturvallisuus, Koneoppiminen, Syväoppiminen

Abstract: Nowadays artificial intelligence is widely used for improving efficiency in our society, but it also poses new risks and vulnerabilities that can be exploited through various attacks. This thesis aims to analyze the most prevalent cybersecurity vulnerabilities related to AI and concludes that the learning process of an AI model is its primary vulnerability, which can be safeguarded through the use of external AI systems. This thesis is conducted as a literature review.

Keywords: Artificial intellligence, Vulnerability, Data poisoning, Cybersecurity, Machine learning, Deep learning

Kuviot

Sisällys

1 Johdanto

Tekoäly on ihmisen älykkyyttä matkiva järjestelmä, jota käytetään isoja data määriä prosessoivana työkaluna. Se mahdollistaa laitteiston älykkään toiminnan, jonka ansiosta tietokoneita pystyy hyödyntämään eri tarkoituksissa, joissa se ei olisi ennen ollut mahdollista. (Vähäkainu ym. 2019) Tekoäly on laajasti implementoitava teknologia, sillä sitä voidaan teoriassa käyttää missä tahansa älykkyyttä vaativassa tehtävässä. Tekoälyä hyödynnetään monipuolisesti esimerkiksi laitteiston ohjaamisessa, ihmiskielen tulkinnassa, kuvantunnistuksessa ja matemaattisten yhtälöiden ratkaisemisessa. (Russell ym. 2021, s.19)

Tekoäly on erittäin nopeasti kehittyvä ala, joka herättää ihmisissä suurta kiinnostusta. Tutkimusten mukaan, sen tiedetään nyt jo tekevän miljardeja dollareita voittoja yrityksille. (Russell ym. 2021, s.19) Tekoälyn hyödyt ovatkin monelle selkeät; tietokoneiden hoitaessa toistuvia yksinkertaisia tehtäviä, ihmisille jää enemmän aikaa. Tämä voi nostaa kansan yleistä hyvinvointia, sekä lisätä innovaatiota tuotteiden ja palveluiden kehityksessä. (Russell ym. 2021, s.49) Tekoälyn avulla voidaan ainakin teoriassa suorittaa kaikki henkeä uhkaavat työt koneilla. Näinkin nopeasti yleistyvä teknologia nostaa kuitenkin huolen siitä, huomioidaanko tekoälystä aiheutuvia riskejä liian vähän.

Tämän tutkielma tarkoituksena on lisätä tietämystä tekoälyn heikkouksista analysoimalla koneoppimismalleihin perustuvia luonnollisia haavoittuvuuksia, sekä tunnetuimpia hyökkäyksiä niitä vastaan. Tutkielmassa myös esitellään valmiiseen tutkimusdataan perustuva yhteenveto tavoista puolustaa tekoälyä edellä mainituilta ilmiöiltä. Seuraavaksi esitetään tekoälyä koneoppimisen näkökulmasta, sekä avataan eri käsitteitä koneoppimismallin sovelluksista. Kolmannessa luvussa tutkitaan tunnettuja tekoälyn haavoittuvuuksia, sekä eri hyökkäyksiä sen oppimista vastaan. Neljännessä luvussa käsitellään tekoälyn vaaroja kriittisissä järjestelmissä, eli tutkitaan epäluotettavan päätöksenteon riskejä yleisesti, sekä tapauskohtaisesti. Tapauskohtaiset tutkimukset ovat konkreettisia esimerkkejä jokapäiväisistä palveluista, jossa tekoäly tulee yleistymään nopeasti lähitulevaisuudessa. Viidennessä luvussa käsitellään eri tutkimuksiin perustuvia puolustautumisstrategioita tekoälyn luonnollisia heikkouksia, sekä yleisimpiä hyökkäyksiä vastaan.

2 Tekoälymallit ja oppiminen

Älykkyys on monitahoinen käsite, jolle ei suoranaisesti ole yhtä tiettyä selitystä. Kuitenkin laajalti hyväksytty tapa kuvailla älykkyyttä on kyky oppia kokemuksista luomalla muistoja ja ymmärrystä. Oppiminen on perusta ajatteluun, johon liittyy ongelmanratkaisutaidot, toistuvuuksien tunnistaminen, sekä kyky päätellä ja tehdä valintoja. (Vähäkainu ym. 2019) Tekoälyn tarkoituksena on mallintaa ihmisille tyypillistä älykkyyttä tietojenkäsittelyjärjestelmiin.

Tekoäly tarkoittaa minkä tahansa laitteiston keinotekoista älykkyyttä, jonka avulla ne pystyvät käsittelemään dynaamisia muuttujia ja reagoimaan niihin automaattisesti haluttujen tavoitteiden mukaan. (Vähäkainu ym. 2019) Jotta järjestelmä toimii kognitiivisesti, on sille mallinnettava älykkyyttään opettamalla. Tämä mahdollistaa tietojenkäsittelyn älykkään automatisoitumisen, sillä järjestelmien ei enää tarvitse seurata tiettyjä syötteitä ja koodattuja sääntöjä, vaan ne pystyvät adaptiivisesti prosessoimaan erilaisia muuttujia ja päättelemään tuloksia opittujen riippuvuuksien pohjalta.

Tämä luku kontekstualisoi tekoälyn rakennetta oleellisesti tutkimusta varten koneoppimismetodien näkökulmasta. Rakenne ja sen monimutkaisuus vaihtelee tehtävän mukaan, mutta lähtökohtaisesti jokainen koneoppimalli prosessoi dataa käyttäen eri oppimisalgoritmeja ja tekee valintoja perustuen sen koulutusvaiheessa tehtyihin ennusteisiin (engl. predictions) ja luokitteluihin (engl. classification). (Ergen 2019) Luvussa avataan koneoppimista ja sen mallien toimintaa, sekä syväoppia ja siihen perustuvia tutkimusaiheeseen liittyviä sovelluksia.

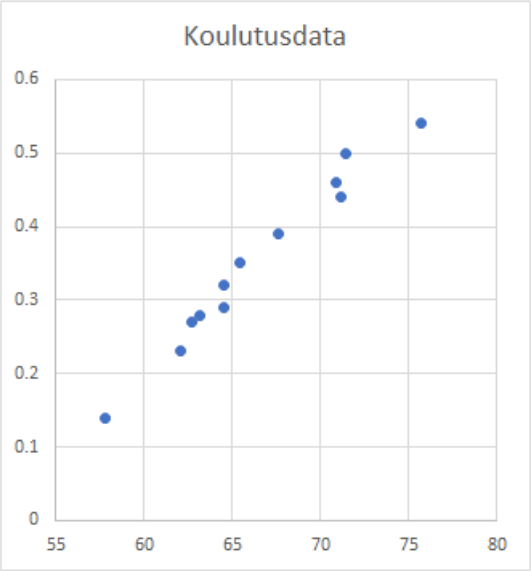
2.1 Koneoppi (machine learning, ML)

Nykypäivänä suurin osa tekoälyn oppimisesta tapahtuu matemaattisien koneoppialgoritmien avulla. Koneoppimallissa järjestelmälle syötetään suuria määriä opetusdataa, josta algoritmit etsivät toistuvia riippuvuuksia tiettyjen syöttöjen ja tulosten välillä, joiden avulla malli kykenee tekemään tarkkoja ennusteita ja luokitteluita käsitellessään uutta dataa. Koulutettua mallia iteroidaan validointi datalla, jonka avulla varmistetaan, että malli käyttäytyy tarkoi-

tetulla tavalla. (Ergen 2019) Koneoppimiselle on eri menetelmiä, joita käytetään erilaisten tekoälyjen rakentamiseen.

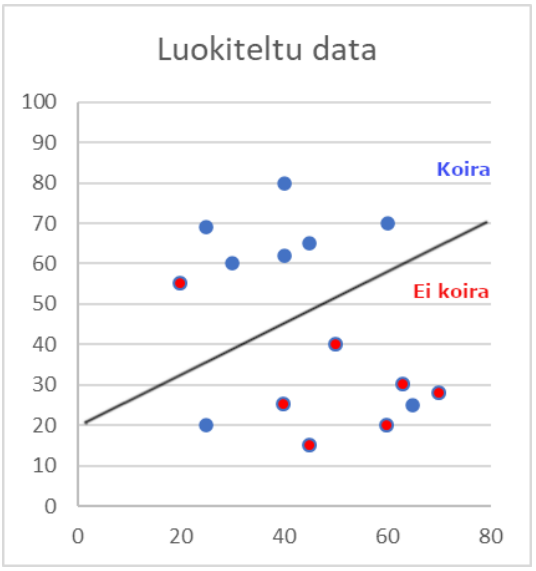
Ohjatussa oppimisessa koneoppimallia koulutetaan datalla, jossa haluttu lopputulos tiedetään. (Russell ym. 2021, s.671) Valmiita tulos muuttujia verrataan oppimismallin tekemiin ennusteisiin ja luokitteluihin. Ohjatun oppimisen malli siis kartoittaa syöttö-tulos riippuvuuksia: $f : X \rightarrow Y$, eli $y = f(x)$, jossa y on tulos, x on syöte ja funktio on kartoittava malli. (Fernandes de Mello ja Antonelli Ponti 2018, s.5) Vaaditun tehtävän mukaan, malli rakennetaan joko luokittelumalliksi, tai ennusteita tekeväksi regressiomalliksi.

x	y
63.16	0.28
64.54	0.29
67.62	0.39
57.81	0.14
71.43	0.5
71.17	0.44
62.73	0.27
65.44	0.35
75.71	0.54
64.47	0.32
62.02	0.23
70.92	0.46



Kuvio 1. Kuvaaja ohjatun oppimisen regressiomallista, perustuu Fernandes ym.(2018) s.7

Kuvaajassa syötetään koneoppimisalgoritmin funktiolle x arvolle ilmanpaine ja y arvolle sateen prosentuaalinen mahdollisuus. Malli rakennetaan ennustamaan sateen mahdollisuutta pelkästään ilmanpaine syötteen avulla, opettamalla sille tämän tapauksen x:n ja y:n riippuvuudet toisistaan. (Fernandes de Mello ja Antonelli Ponti 2018)



Kuvio 2. Kuvaaja ohjatun oppimisen luokittelumallista, perustuu Skansi(2018) s.55

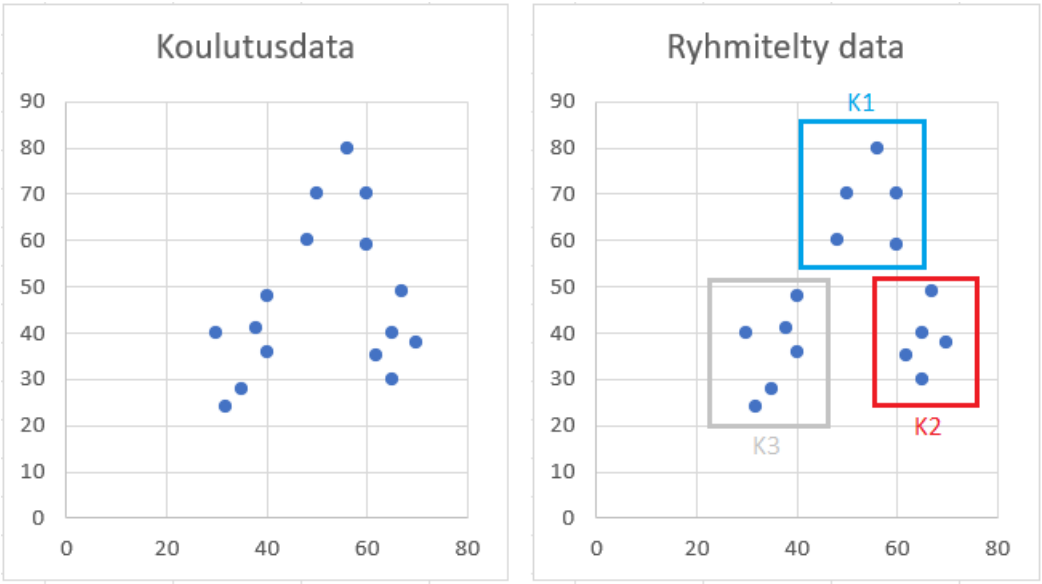
Luokitteleva malli koulutetaan opetusdatalla, joka sisältää eri tuloksellisia esimerkkejä, jonka avulla uudet syötteet luokitellaan niiden ominaisuuksien mukaan. Esimerkiksi jos mallin tarkoituksena olisi tunnistaa, sisältääkö syötteen kuva koiran, malli luokittelisi kuvat joko "koirana", tai "ei-koirana". Luokittelu tehdään vertaamalla syötedatan kuvien ominaisuuksia koulutusdatan kuviin, joiden tulos tiedetään. Malli pystyy näin arvioimaan mihin uusi syöte luokitellaan. (Skansi 2018) Kuvaajassa viiva erottaa mallin valinta-alueet (engl. decision space), joka näyttää graafisesti mitkä ominaisuudet kullakin luokalla on. Valinta-alueet siis muodostuvat koulutusdatan ominaisuuksien pohjalta. Kuten kuvaajasta nähdään, luokittelut eivät aina ole oikein, jonka takia onkin tärkeää validoida mallin toimintaa tutkimalla sen luotettavuutta. Skansin (2018) mukaan, mallin true- ja false-positive, sekä true- ja false-negative arvoilla voidaan tutkia sen toimintaa. Tässä tapauksessa arvot siis kertovat, kuinka monta oikeaa ja väärää "koira", sekä montako oikeaa ja väärä "ei-koira"luokittelua malli on tehnyt. Näiden arvojen avulla voidaan laskea mm. mallin tärkein ominaisuus, eli prosentuaalinen tarkkuus:

$$tarkkuus = \frac{truePositive + trueNegative}{syotteidenMaara}$$

Ohjattua oppimista käytetään tilanteissa, joissa halutaan tekoälyn antavan täsmällisiä tuloksia, tekemällä opittuun perustuvia oletuksia uuden datan kanssa. Esimerkiksi roskapostin havaitsemiseen tarkoitetun tekoälyn mallin rakentamiseen käytetään ohjattua oppimista. (Shar-

ma, Kaur ja Semwal 2022)

Ohjaamattomassa oppimisessa koneoppimisalgoritmille annetaan vain syötettävää dataa. Mallia käytetään silloin, kun datan lopputuloksia ei verrata valmiisiin muuttujiin, vaan sitä käytetään samalla koulutusdatan tutkimiseen. (Sharma, Kaur ja Semwal 2022) Toisin sanoen algoritmien annetaan toimia itsekseen, jotta ne löytävät uusia riippuvuuksia suuresta määrästä dataa ilman ihmisen vuorovaikutusta. Ohjaamattomassa oppimisessa käytetyt algoritmit iteratiivisesti ryhmittelevät syötteitä X niiden ominaisuuksien ja piirteiden mukaan. Esimerkiksi K-means-algoritmi on yksi yleisimmin käytetyistä ryhmittelyalgoritmeista, jossa valitaan tietty määrä K-pisteitä. Jokainen syöte X määritellään tietylle K-pisteelle samankaltaisuuksien mukaan. Ryhmittymien avulla voidaan tutkia syötteiden ominaisuuksista johtuvia toistuvuuksia (Chinnamgari 2019, s.16)

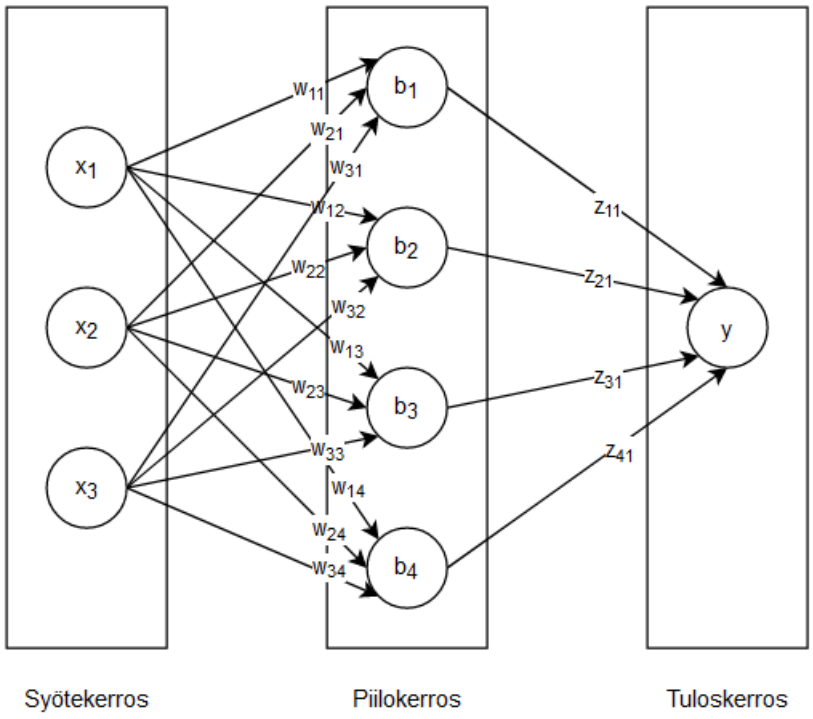


Kuvio 3. Esimerkki K-means ryhmittelystä, perustuu Sunil Kumar(2019) s.16

Vahvistusoppimisessa malli saa toimintansa jälkeen palautteen vain siitä, onko sen toiminnasta seurannut oikea vai väärä tulos. (Russell ym. 2021, s.671) Malli joutuu niin sanotusti "trial-and-error-metodilla selvittämään, mitä toimintoja on muutettava, jotta saavutetaan positiivinen palaute. (Ergen 2019)

2.2 Syväoppi (deep learning, DL)

Syväoppiminen on koneoppimisen osajoukko, jonka periaatteena on mallintaa laitteiston tekoälyä keinotekoisella neuroverkostolla. (Russell ym. 2021, s.801) Sen tarkoituksena on simuloida ihmisen aivojen toimintaa, jonka monessa eri kerroksessa sijaitsevat neuronit prosessoivat syötettä ennen tuloksen välittämistä seuraavalle kerrokselle. Datan liikkuessa kerroksesta toiseen, sen prosessointi monimutkaistuu. Viimeisen kerroksen tulos on mallin ennuste tai luokittelu. Neuroverkon käsitellessä koulutusdataa, jokainen neuroni päivittyy tarvittaessa, jotta koko verkon tulos saadaan halutuksi. (Ergen 2019) Kuten koneoppimisessa, myös neuroverkostojen funktio on $y = f(x)$, mutta malli on rakenteeltaan monikerroksinen, jossa syöte on: $x = (x_1 + x_2 + x_3 ... + x_n)$, jolloin tulos on $y = (y_1 + y_2 + y_3 ... + y_m)$. (Skansi 2018, s.81)



Kuvio 4. Esimerkki kuvaus neuroverkoston rakenteesta, perustuu Skansi(2018) s.80

Skansin (2018) mukaan neuroverkosto koostuu syöte-, piilo ja tuloskerroksesta. Jokainen syöte siirtyy piilokerroksen neuroneihin käsiteltäviksi niiden välisiä yhteyksiä pitkin. Jokaisessa neuronin välisessä yhteydessä on muuttuja w, eli paino (engl. weight), jonka tarkoi-

tuksena on muokata syötettä sen liikkuessa neuronien ja kerroksien välillä. Piilokerroksen neuroneilla on mukautuva vakiotermi b, joka myös lisätään syötearvoon neuronin käsittelyssä:

$z_{11} = w_{11} * x_1 + w_{21} * x_2 + w_{31} * x_3 + b_1$

Piilokerroksen neuronien summat viedään aktivointifunktiolle, jonka avulla saadaan tulos

$y = \sigma(z_{11}) + \sigma(z_{21}) + \sigma(z_{31}) + \sigma(z_{41})$, jossa σ on aktivointifunktio.

Skansin ym. (2018) mukaan painon ja vakiotermien arvot ovat lähtökohtaisesti satunnaisesti valittuja, mutta arvojen tarkoituksena on iteratiivisesti mukautua niin, että tulos olisi mahdollisimman lähellä haluttua. Neuroverkoston tuloksia siis verrataan aktiivisesti koulutusdatan tuloksiin. Tämä toteutetaan esimerkiksi yleisesti käytetyllä "mean squared error", eli MSE-virhefunktiolla, jossa y on neuroverkoston tulos ja y_t on koulutusdatan haluttu tulos:

$MSE = \frac{1}{2} * (y - y_t)^2$

Iteraation tarkoituksena on minimoida virhefunktion tulos automaattisesti muuttamalla neuroverkostossa esiinty arvoja. Tämän takia mallin rakentaminen saattaa kestää vaihtelevasti jopa viikkoja. Yksinkertaisemmin toimivien koneoppimallien koulutusvaiheet ovat huomattavasti nopeampia kuin neuroverkostojen käyttö. Syväoppimismallin etuna on kuitenkin korkea laatuisemmat, sekä tarkemmat tulokset mallin toiminnassa. (Ergen 2019)

2.3 Luonnollisen kielen prosessointi (natural language processing, NLP)

Luonnollisen kielen prosessointi on koneoppimisen osa-alue, jonka avulla tekoäly kykenee ymmärtämään ihmisten luonnollisilla kieliä. (Deng ja Liu 2018, s.1) NLP:n tarkoitus on muotoilla ihmiskielisiä syötteitä tekoälylle ymmärrettäväksi, sekä generoida tekoälyn tulosteista ihmiskielisiä. NLP-sovelluksia ovat esimerkiksi puhetunnistus, konekääntäminen, sekä tunneanalyysi. (Deng ja Liu 2018, s.1)

NLP perustuu koneoppimallien hyödyntämiseen. Perinteiset mallit ja niiden algoritmit eivät kuitenkaan ole tarpeeksi tehokkaita saavuttaakseen täydellisen ihmiskieliosaamisen tekoälyissä. Nykypäivänä yleistyneet syväoppimallit ja niissä käytetyt neuroverkostot ovat aiheut-

taneet suuria harppauksia NLP-teknologian kehityksessä. (Deng ja Liu 2018, s.6) NLP on yksi merkittävimmistä tekijöistä, joka on edistänyt tekoälyn suosion kasvua peruskäyttäjien keskuudessa. Esimerkiksi tehokkaasti valtavirtaistunut Chat-GPT on neuroverkostoista rakennettu kielimalli, joka NLP-algoritmien avulla käsittelee käyttäjien pyyntöjä ja luo niiden pohjalta ihmiskielisiä vastauksia. (OpenAI 2021)

Nopeasta kehityksestä huolimatta, NLP:ä on edelleen suuria haasteita johtuen ihmiskielen monimutkaisuudesta. Vaikka NLP mallit antavat jo hyviä tuloksia, ne voivat silti usein olla epäluotettavia, koska useimmat nykypäivän koneoppimallit eivät omaa päättelykykyä, eivätkä osaa toimia oikein odottamattomissa tapauksissa. (Deng ja Liu 2018, s.12)

2.4 Konenäkö (computer vision)

Konenäkö on tietojenkäsittelyn osa-alue, jonka avulla kyberfyysinen-järjestelmä pystyy analysoimaan ja tunnistamaan visuaalista dataa, kuten esimerkiksi ympäristöä, QR-koodeja tai ihmisten kasvoja tietokonealgoritmien avulla. (Dadhich 2018, s.7) Nykypäivänä konenäkö perustuu lähinnä syväoppialgoritmien muodostamiin luokitteluihin kuvia sisältävästä opetusdatasta. (Dadhich 2018, s.11) Koulutettu malli tunnistaa järjestelmän havaitsemia asioita vertaamalla niitä opetusdatan kuviin. Konenäköön liittyy myös erilaisia menetelmiä, joita tekoäly voi käyttää ympäristön tutkimisessa.

Hahmontunnistus (engl. object detection) on yksi tärkeimmistä konenäön ominaisuuksista, jota käytetään monissa eri järjestelmissä, kuten robotiikassa, autonomisissa ajoneuvoissa, sekä valvontalaitteistossa. Konenäkö tunnistaa havaitsemiaan hahmoja luokittelemalla esimerkiksi kameran, tai muiden sensoreiden keräämää dataa. (Dadhich 2018, s.11) Hahmonseuraus (engl. object tracking) on konenäön kyky seurata esimerkiksi hahmon liikettä kuvien sarjaa tutkiessa. (Dadhich 2018) Hahmonseurauksen avulla tekoäly pystyy seuraamaan liikkuvia kohteita. Konenäkö hyödyntää myös kuvan käsittelyyn liittyviä teknologioita. Tekoäly pystyy tutkimaan kuvasta hahmojen syvyyksiä prosessoimalla ne kolmiuloitteisena rakenteena, segmentoimaan saman hahmon pikselit ryhmiin, sekä generoimaan uusia kokonaisuuksia olemassa olevien kuvien hahmoista. (Dadhich 2018)

3 Tekoälyn tunnetut haavoittuvuudet

Kuten minkä tahansa tietojenkäsittelyjärjestelmän, myös tekoälyn on ylläpidettävä luottamuksellisuutta, palvelun saatavuutta, sekä tiedon eheyttä. (Eggers ja Sample 2020) Optimaalisessa tilanteessa älykkäiden laitteiden käyttö kriittisissä ympäristöissä lisää turvallisuutta ja tehokkuutta perinteisiin autonomisiin järjestelmiin verrattuna. (Terziyan, Golovianko ja Gryshko 2020)

Tekoälyn yleistyessä eri järjestelmissä, väärinkäytön potentiaali kasvaa huomattavasti. Älykkyyden implementointi eri aloille tietämättä sen toiminnasta ja riskeistä saattaa olla pahimmassa tapauksessa erittäin vaarallisia, sillä tekoäly on vielä kaukana täydestä luotettavuudesta. Tekoälyyn liittyy useita rakenteellisia ongelmia, jotka voivat aiheuttaa odottamattomia tilanteita mallien päätöksenteossa. Usein puhutaan tekoälyn käytöstä hyökkäyksen tai puolustuksen vektorina, mutta järjestelmä voi myös itsessään joutua vaikuttamisen uhriksi. Hyökkäys voi siis tapahtua suoraan tekoälyn sovelluksiin ja oppimismetodeihin. Pahimmassa tapauksessa hyökkääjät voivat vaikuttaa tekoälyn koulutusdataan, joka muuttaa mallin toimintaa täysin. (Eggers ja Sample 2020)

Tässä luvussa käydään läpi haavoittuvuuksia, jotka tiedetään olevan vaaraksi tekoälyjärjestelmille. Luvussa käsitellään sen rakenteeseen perustuvia heikkouksia, sekä tutkitaan eri hyökkäyksiä, jotka kohdistetaan tekoälyn oppimismallien toimintaa vastaan.

3.1 Haasteet tekoälyn rakenteessa

Tekoälyteknologiassa on tapahtunut huimaa kehitystä viime vuosina muun muassa datan saatavuuden ja syväoppimismetodien ansiosta. Nykypäivän tekoäly järjestelmät pärjäävät vaikuttavasti tietyissä tehtävissä, kuten hahmotunnistuksessa tai ihmiskielen prosessoinnissa, mutta kohtaavat silti ongelmia tilanteissa, jotka vaativat niin sanotusti "maalaisjärkeä"(engl. common sense). Tämä tarkoittaa sitä, että tekoäly ei pysty intuitiivisesti selvittämään odottamattomia tapauksia, jotka vaativat laajaa ymmärrystä fyysisen- ja sosiaalisen maailman toiminnasta. Tekoälyllä on siis vaikeuksia hahmottaa kontekstia, esimerkiksi kielien kääntäjä järjestelmä ei välttämättä näe sanojen kulttuurisia nyansseja. Tällaisen "maalaisjärki-

sen"tekoälyn saavuttamiseksi, lähestymistavat sen rakentamiseen vaatisivat huomattavaa uudelleen suunnittelua. (Choi 2022)

Edellä mainittuun haasteeseen perustuu myös ongelmia tekoälyn luotettavuudessa, jotka liittyvät järjestelmän robustisuuteen (engl. robustness). Robusti käsitteenä voi tekoäly kontekstissa tarkoittaa lukuisia asioita, mutta tässä tutkielmassa sillä kuvataan koneoppimismallien kykyä reagoida luotettavalla tavalla ristiriitaiseen, tai odottamattomaan syötedataan. (Chen ja Das 2023) Nykyaikaisten mallien "lack of robustness", eli robustisuuden puute, voi aiheuttaa luonnostaan vääriä ennusteita ja luokitteluita tekoälyn toiminnassa. Virhe tuloksia voi esiintyä esimerkiksi hahmotunnistuksessa muuttuvissa sääolosuhteissa, mutta robustisuuden puute altistaa mallin myös ulkopuoliselle vaikutukselle. (Eykholt ym. 2018)

Robustisuuden kannalta olisi siis tärkeää, että tekoälymallien toiminnan yksityiskohtia pidettäisiin käyttäjille selkeänä. Nämä niin sanotut white-box mallit mahdollistavat täydellisen järjestelmän testaamisen virheiden ja hyökkäyksien varalta. (Chen ja Das 2023) Huomattava mallien kehitys ja monimutkaistuminen ovat kuitenkin vähentäneet tekoälyjärjestelmien läpinäkyvyyttä ja luotettavuutta. Esimerkiksi neuroverkostot ovat niin sanottuja black-boxeja, sillä ne muodostuvat suurista määristä neuroneita, joiden vakiotermit ja yhteyksien painot muuttuvat automaattisesti mallin koulutusvaiheessa. Syväoppimismallien toimintaa ja päätöksentekoa on siis hankala tulkita ja ymmärtää, sillä ihmisen vuorovaikutus mallin rakentumiseen on vähäistä. Chenin ym. (2023) mukaan tietämättömyys siitä, kuinka järjestelmä on saapunut päätökseen, voi olla vaarallista vastuullisia valintoja tehdessä esimerkiksi terveydenhuollossa. Onnistuneet hyökkäykset black-box-mallia vastaan ovat huomattavasti vaarallisempia, sillä vihamielisesti manipuloitua järjestelmää ei välttämättä huomata.

3.2 Tekoälyn ennakkoasennoituminen (A.I bias)

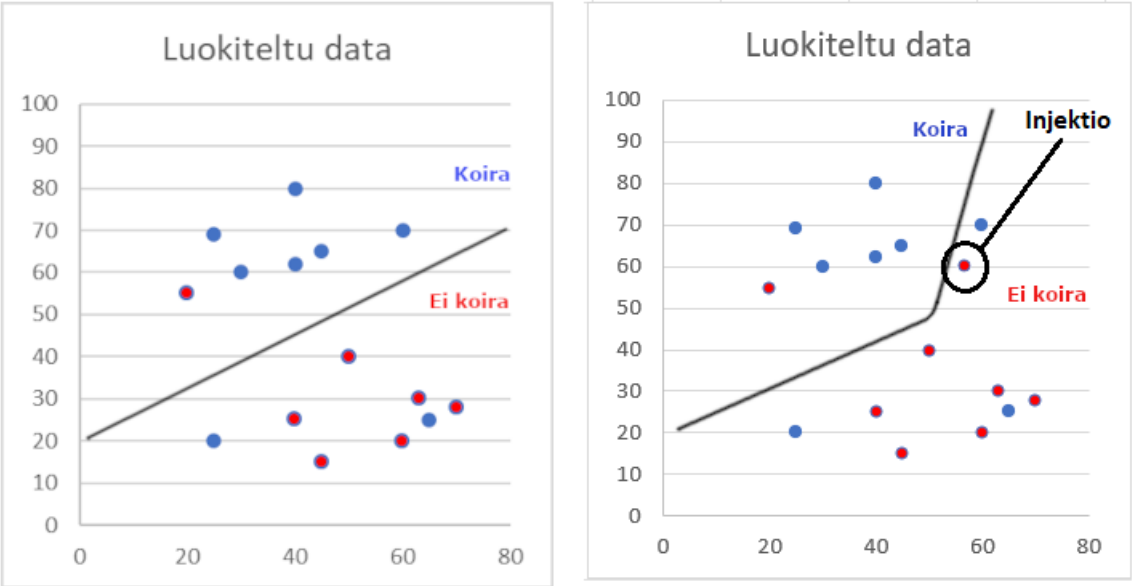
Tekoälyn tehokkuus riippuu täysin sen koulutusdatasta. Jotta tekoälyjärjestelmä toimisi oikein, koulutusdatan pitäisi olla korkealaatuista ja tarkoituksenmukaista, muuten koulutettu mallin tulee poikkeamaan halutusta ja toimimaan odottamattomalla tavalla. Myös validointidatan puutteellisuus voi aiheuttaa mallin väärin rakentumista. (Eggers ja Sample 2020) Toisin sanoen mallista voi tulla ennakkoasennoitunut sen koulutusdatan mukaisesti.

Tekoälyn ennakkoasennoitumista tapahtuu luonnostaan esimerkiksi huonosti valitun koulutusdatan takia. Internet on täynnä harhauttavaa dataa, jonka seurauksen tekoäly saattaa tiedostamatta rakentua vääränlaiseksi. On tärkeää huomata, että joissakin tilanteissa tietyistä demografisista piirteistä kerääntyy vähemmän koulutusdataa, mikä voi vaikuttaa tekoälyn suorituskykyyn näillä alueilla. Tällaisia "harvinaisempia"piirteitä sisältävää koulutusdataa tulisi siis painottaa erityisesti kontekstista riippuen, kuten sukupuolitasa-arvoa korostaessa rekrytointimallia rakentaessa. Mallin tarkasta validoinnista huolimatta, se voi joutua hyökkäyksen kohteeksi, jossa ilkeämieliset vaikuttajat yrittävät tahallisesti korruptoida koulutusdataa, jotta tekoälystä tulisi ennakkoasenteinen. (Eggers ja Sample 2020) Ennakkoasentuitunut tekoälyjärjestelmä voi ilmentyä oikeassa maailmassa esimerkiksi tiettyjen väestöryhmien syrjintänä. (Choi 2022)

3.3 Tiedonmyrkytys (Data poisoning)

Hyökkäys, jossa vihamielinen vaikuttaja yrittää muokata koneoppimallin koulutusalgoritmia, tai -dataa, kutsutaan tiedonmyrkyttämiseksi (engl. data poisoning). Myrkyttämisen tarkoituksena on muokata koneoppimallin toimintaan liittyvää dataa, jotta sen ennusteet ja luokittelut vääristyvät ja muuttuvat epäluotettaviksi. (Hu ja Hu 2020) Monipuoliset tiedonmyrkyttämishyökkäykset ovat yksi suurimmista kyberturvallisuuden heikkouksista, mitä koneoppi- ja syväoppimalleilla on. (Terziyan, Golovianko ja Gryshko 2020)

Lähtökohtaisesti myrkytyshyökkäykset vaativat sen toimeenpanijalta jonkinlaista tietämystä kohdemallin rakenteesta.(Chen ja Das 2023) Hyökkäys voi olla kohdennettu korruptoimaan tiettyä ominaisuutta mallin toiminnassa, tai kohdistamaton, jonka tarkoituksena on muokata mitä tahansa se pystyy. (Hu ja Hu 2020) Hyökkäyksen toimenpiteisiin vaikuttaa saatavan tiedon määrä, eli onko malli arkkitehtuuriltaan black-box, white-box, tai niiden välimuotoa. Tämän pohjalta myrkytys voi olla logiikan korruptointia, jossa oppimisalgoritmeja muokataan, datan manipulointia, jossa koulutusdataa muokataan, tai datan injektointia, jossa koulutusdataan lisätään hyökkääjän muuttujia. (Hu ja Hu 2020)



Kuvio 5. Esimerkki data-poisoning injektiosta, perustuu Terziyan ym. (2020)

Riippuen siis myrkytyshyökkäyksestä, väärennettyä dataa voidaan lisätä, ja olemassa olevaa dataa voidaan muokata tai poistaa. Vihamielisten koulutusnäytteiden seurauksena koneoppimismallin valinta-alueet (engl. decision space) vääristyvät, jolloin malli rakentuu epäluotettavaksi. (Terziyan, Golovianko ja Gryshko 2020) Kuvaajasta näemme, että injektoitu data luo niin sanotun myrkytetyn alueen, jossa tiettyjen ominaisuuksien syötteet luokittuvat väärin.

3.4 Adversariaali hyökkäykset (Adversarial attacks)

Adversariaali hyökkäykset (engl. adversarial attacks) ovat koneoppiin kohdistettua vihamielistä vaikuttamista, ja kuten myrkytyshyökkäyksissä, niiden tarkoituksena on manipuloida mallin toimintaa. Nämä hyökkäykset kuitenkin eroavat luonteeltaan, sillä niiden toiminta perustuu jo koulutetun mallin syötteeseen ja tulosteeseen. (Finlayson ym. 2019)

Mallin väistö (engl. model evasion) on hyökkäys, jossa käytetään vihamielisiä esimerkkejä syötteinä, jotta mallin ennuste tai luokittelu olisi väärä. (Eggers ja Sample 2020) Finlaysonin ym. (2019) mukaan syötteitä, jotka ovat tarkoituksella muodostettu sekoittamaan tai hämäämään koneoppimismallin toimintaa aiheuttamalla odottamattoman tilanteen, kutsutaan vihamielisiksi esimerkeiksi (engl. adversarial example). Tällainen syöte voisi yksinkertaisimmillaan olla esimerkiksi estetyn roskapostin muuttaminen lisäämällä, tai muokkaamalla sanoja,

jotta malli ei enää tunnistaisi sitä roskapostiksi. Tutkimuksissa on huomattu, että vihamielisiä esimerkkejä kohdistuu laajalti jokaiselle koneoppimisen tyypille, kuten mm. konenäölle ja luonnollisen kielen prosessoinnille. Adversariaali hyökkäyksillä on erilaisia lähestymistekniikoita, joiden valinta perustuu hyökkääjään tavoitteisiin, sekä hyökättävän mallin arkkitehtuuriin.

Mallin varastaminen (engl. model stealing) on adversariaali hyökkäämisen tekniikka, joka perustuu mallin toiminnan tietovuotoihin. Hyökkääjä iteratiivisesti syöttää mallille muuttuvaa dataa, jota verrataan mallin antamiin tuloksiin. Syöte-tulos relaatiolla, hyökkääjä pystyy kartoittamaan black-box mallin sisäistä toimintaa. Hyökkäyksen tarkoituksena on kouluttaa klooni kyseisestä mallista, jonka avulla sen toimeenpanija voi joko kehittää kloonistaan paremman kuin alkuperäinen, tai käyttää sitä vihamielisten esimerkkien luomiseen, joita voidaan käyttää uusissa hyökkäyksissä. (Juuti ym. 2019)

Mallin inversio (engl. model inversion) on idealtaan saman kaltainen kuin mallin varastaminen. Inversion tarkoituksena on kuitenkin jälleenrakentaa mallien koulutusdataa sen tulosteiden pohjalta. (Wang ym. 2022) Myös inversio siis perustuu mallin toiminnan tutkimiseen syöte-tulos riippuvuuksien avulla. Koulutusdatan selviäminen on erittäin suuri kyberturvallisuusriski, sillä koneoppialgoritmit saattavat käyttää erittäin yksityistä dataa mallien rakentamiseen. (Wang ym. 2022)

4 Tekoälyn vaarat kriittisissä järjestelmissä

Aiemmasta luvusta käy ilmi, kuinka haavoittuvainen tekoäly ainakin toistaiseksi vielä on. Kaikista hyödyistään huolimatta, älykkään automaation käyttöönotto kriittisissä järjestelmissä aiheuttaa uusia odottamattomia haavoittuvuuksia niihin liittyvissä järjestelmissä ja niin vaaroja ihmisille, sekä ympäristölle. Tekoäly vaatisikin huomattavaa kehitystä, mutta monet käytössä olevat järjestelmät ovat riippuvaisia keinotekoisesta älykkyydestä.

Tässä luvussa konkretisoidaan tekoälyn haavoittuvuuksista johtuvia vaaratilanteita korkean vastuun järjestelmissä. Luvussa tutkitaan edellisessä luvussa käsiteltyjä luonnollisia tekoälyn rakenteeseen, sekä vihamieliseen vaikuttamiseen perustuvia riskejä yleisenä ilmiönä ja tapauskohtaisesti eri aloilla.

4.1 Riskit tekoälyn päätöksenteossa

Keinotekoisen älykkyyden tarkoituksena on dynaamisesti automatisoida kyber-fyysisien laitteiden toimintaa, jotta ne eivät vaatisi ihmisen vuorovaikutusta. Tähän liittyy niin suurien data määrien käsittelyä, kuin uuden sisällön generointiakin. (Terziyan, Golovianko ja Gryshko 2020) Tekoälyn päätöksenteossa on luonnostaan riskitekijöitä, mutta se on myös haavoittuvainen kyberhyökkäyksiä vastaan. Keinotekoisen älykkyyden kognitiiviset toimintavirheet saattavat siis aiheuttaa pahimmassa tapauksessa hengenvaarallisia tilanteita.

Väärät luokittelut ja ennusteet voivat olla seurausta järjestelmävirheestä, koneoppimismallin huonosta koulutusdatasta, tai ulkopuolisesta vaikuttamisesta. Hyökkäysten tarkoituksena onkin aiheuttaa luotettavalle tekoälylle tilanteita, jossa sen päätöksenteko toimii väärin. Tästä robustisuuden puutteesta johtuva epäluotettavuus ei ole suotavaa korkean vastuun ympäristöissä, kuten esimerkiksi autonomisissa ajoneuvoissa, energiantuotannossa ja -jakelussa, tai terveydenhuollossa. (Finlayson ym. 2019)

4.1.1 Tapaus: robottiautot

Robottiikka tieliikenteessä on luultavasti yksi ensimmäisistä kriittisen ympäristön tekoälyistä, joka monella tulee mieleen, sillä henkilövahinkoja voi seurata jo pienimmästäkin virheestä. Robottiauto on ajoneuvo, joka kykenee analysoimaan muuttuvaa ympäristöä ja reagoimaan erilaisiin tilanteisiin automaattisesti. Robottiautojen älykäs toiminta perustuu syväoppimisella rakennettuihin neuroverkostoihin, sekä niistä sovellettuun konenäön hyödyntämiseen. (Eykholt ym. 2018) Robottiauto tutkii ympäristöään kameroilla, sekä erilaisilla sensoreilla ja analysoi havaintojaan hahmontunnistuksen ja -seurauksen avulla. Auton älykkyyden robustisuuteen liittyvät ongelmat voivat kuitenkin haitata sen toimintakykyä, jos konenäkö havaitsee jotain odottamatonta, tai neuroverkostoa vastaan hyökätään. (Eykholt ym. 2018)

Robottiautojen tekoälyä vastaan voidaan hyökätä digitaalisen tason lisäksi myös fyysisessä ympäristössä. Konenäön hahmontunnistusta vastaan voi pienellä kynnyksellä aiheuttaa vääriä luokitteluita muokkaamalla liikenneympäristön fyysisiä esineitä. Harmittomalta vaikuttava vandalismi, tai ihmissilmälle vaikeasti havaittavat muokkaukset konenäön kohteissa saattavat aiheuttaa väärän luokittelun ja päätöksen robottiauton neuroverkostossa. Jos konenäkö havaitsee esimerkiksi spraymaalilla vandalisoidun stop-merkin nopeusrajoituksena, tekoälyn väärä päätöksenteko voi olla tuhoisa. (Eykholt ym. 2018)

Digitaalisessa ympäristössä hyökkäykset vihamielisillä esimerkeillä onnistuvat ainakin white-box rakenteisia malleja vastaan. White-box rakenteiset tekoälyt voivat joutua mallin varastamisen kohteeksi, jolloin hyökkääjä saa tiedot robottiauton älyllisen järjestelmän toiminnasta. (Eykholt ym. 2018) Hyökkäykset vaikuttavat välittömästi tekoälyn luotettavuuteen, sillä tietovuodot rakenteesta ja sen mallien toiminnasta voi altistaa muun muassa myrkytyshyökkäyksille. Kyseisellä hyökkäyksellä mallien koulutusdataa voidaan muokata niin, että robottiauton tekoäly tekee vääriä päätöksiä. Tämä onnistuu esimerkiksi lisäämällä koulutusdatan kuviin ihmissilmälle näkymätöntä kohinaa, joka kuitenkin aiheuttaa väärän päätöksen järjestelmässä. (Eggers ja Sample 2020)

4.1.2 Tapaus: terveydenhuolto

Koneoppimismalleihin perustuvien tekoälyjärjestelmien käyttö on yleistymässä myös terveydenhuollossa. Finlaysonin ym.(2019) mukaan vuonna 2018 Yhdysvaltain elintarvike- ja lääkevirasto hyväksyi ensimmäisen autonomisen diagnoosijärjestelmä käyttöönoton, sekä osoitti kiinnostusta kyseisen teknologian yleistämiseen alalla. Tällä hetkellä tekoälyä hyödynnetään eniten terveysalan talousasioissa, kuten vakuutusasioiden tarkastamisessa, mutta sitä halutaan yleistää myös hoitoon liittyvässä päätöksenteossa.

Niin kuin aiemmin tutkielmassa on mainittu, järjestelmän päätöksenteon läpinäkymättömyys voi aiheuttaa tilanteita, jossa muun muassa vääriä diagnooseja voidaan hyväksyä tiedostamatta. Järjestelmän puutteista ja vihamielisestä vaikuttamisesta pohjustuva epäluotettavuus tekoälyä kohtaan luovat heikon perustan tulevaisuuden robotisoituun hoitoon. Esimerkiksi Finlaysonin ym. (2019) mukaan erittäin tarkkoja konenäkölaitteistoja vastaan onnistuttiin käyttämään vihamielisiä esimerkkejä. Niin kuin robottiautot-tapauksessa, hahmontunnistuksen sekoittaminen on mahdollista, joka voi tässä tapauksessa aiheuttaa esimerkiksi pahalaatuisen melanooman luokittelun normaaliksi luomeksi. (Finlayson ym. 2019)

Kriittisiä virheitä, jotka johtuvat hyökkäyksistä, ei terveydenhuollon tekoäly kontekstissa ole vielä havaittu. Finlayson ym. (2019) mukaan matalatasoisempi vaikuttaminen on kuitenkin erittäin yleistä. Esimerkkinä tutkijat onnistuivat hyökkäämään hypoteettista opioidiriski-järjestelmää vastaan hyväksikäyttäen sen NLP-algoritmeja. Järjestelmän tarkoituksena oli kartoittaa terveyshistorian mukaisesti potilaan opioidien väärinkäyttö riskiä. Hyödyntäen hyökkäyksessä synonyymejä terveyshistorian kuvauksessa, onnistuttiin opioidien väärinkäyttäjän riskialttius laskea matalaksi käytännössä samalla potilaskuvauksella. Nämä todistavat sen, että kriittisetkin hyökkäykset ovat mahdollisia terveydenhuollon tekoälyjä vastaan. Esimerkiksi tekoäly voidaan hyökkäyksillä manipuloida syrjimään tiettyjä ihmisryhmiä. (Finlayson ym. 2019)

4.1.3 Tapaus: energiatuotanto

Tekoälyä hyödynnetään erilaisissa energiantuotanto järjestelmissä, kuten älykkäissä mittaristoissa ja -sähköverkoissa, sekä niiden vuorovaikutuksessa internetin kanssa. (Ahmad ym. 2022) Nämä järjestelmät auttavat mm. energian hallinnassa, -tehokkuudessa ja -monitoroinnissa. Tekoälyn avulla tuetaan energiajärjestelmien vakautta, joka varmistaa niiden turvallisen toiminnan. Ahmadin ym.(2020) mukaan tekoälyn syvien neuroverkostojen ansiosta energiajärjestelmät voivat käsitellä suuria määriä dataa, jolloin tuotannon toiminta mukautuu automaattisesti vaatimuksien ja ympäristömuuttujien pohjalta. Tekoälyyn perustuviin turvajärjestelmien avulla voidaan myös havaita esimerkiksi sisäisiä vaaroja, sekä suojata tuotannon tietojenkäsittelyä. Tekoälyn avulla voidaan suojata myös tuotantoympäristöä esimerkiksi hyödyntämällä sitä kameroissa, tai muissa järjestelmissä. (Eggers ja Sample 2020)

Tekoälyn lisääminen energiantuotantoon ja -jakeluun voi tuottaa haasteita. Näitä ovat muun muassa tekoälyn kommunikointiverkon turvallisuus, vanhojen järjestelmien yhteensopivuus, sekä mahdolliset häiriöt älykkäiden järjestelmien toiminnassa. (Ahmad ym. 2022) Lähtökohtaisesti tekoälyn implementoinnissa ollaan onnistuttu, sillä se on olennainen osa nykyajan energiatuotantoa. Esimerkiksi Eggersin ja Samplen (2020) mukaan useissa ydinvoimaloissa tekoälysovelluksia käytetään kyberturvallisuudessa, tuotannon operaatioissa, sekä sisäisten ongelmien havainnoinnissa.

Vaikka turvallista energiantuotantoa ajava järjestelmä olisi erittäin luotettava, voi sen robustisuus kuitenkin kärsiä vihamielisten hyökkäyksien takia. Onnistuneet myrkytys-, tai adversariaali hyökkäykset kyberturvallisuusjärjestelmää vastaan voi altistaa voimalan sisäiset verkostot perinteisemmille hyökkäyksille ilman ulkopuolisen vaikuttajan kiinni jäämistä. Hyökkäykset tuotannossa käytettyä tekoälyä vastaan voi johtaa odottamattomiin laitteistohäiriöihin mallien tekemien väärien päätöksien takia. Edellä mainituilla hyökkäyksillä ulkopuolinen vaikuttaja pystyy esimerkiksi varastamaan tärkeää dataa, tai aiheuttaa voimalan sisäistä sabotaasia, joka voi vaikuttaa kansan turvallisuuteen ja terveyteen, sekä aiheuttaa ylimääräisiä kustannuksia. (Eggers ja Sample 2020)

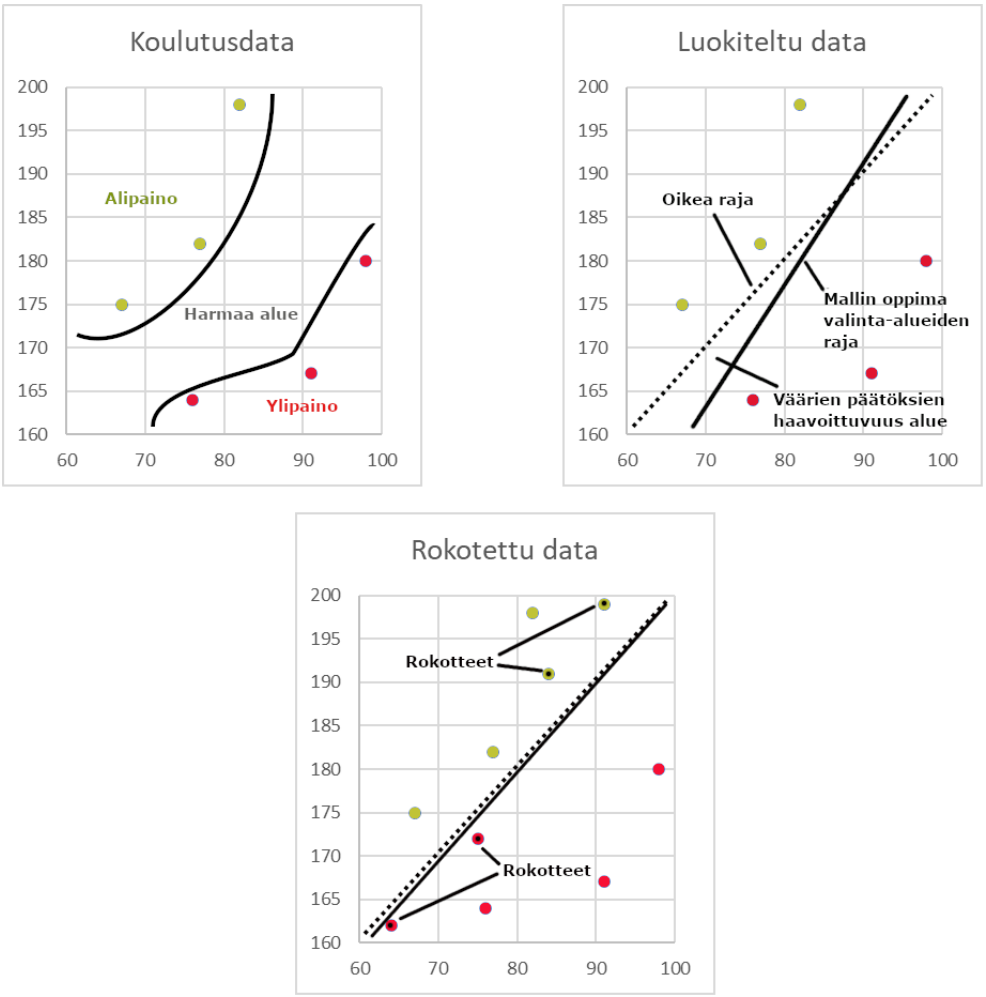
5 Haavoittuvuuksilta puolustautuminen

Nykypäivänä älykkyydestä on kasvanut kyber-fyysisten laitteistojen suurin haavoittuvuus, jonka takia resursseja tulisi keskittää niiden turvallisuuden kehittämiseen. (Terziyan, Golovianko ja Gryshko 2020) Tähän kyberturvallisuus ongelmaan ei kuitenkaan ole helppoa ratkaisua, sillä malleissa on monia erilaisia rakenteisiin ja algoritmeihin liittyviä eroavaisuuksia. (Eggers ja Sample 2020) Useimmissa tekoälyn heikkouksia käsittelevissä tutkimuksissa mainitaan, kuinka oppimismallit ovat älykkyyden haavoittuvin osa-alue ja vaativat turvallisuutta painostavaa kehitystä, jotta saavutetaan täydellinen robustisuus luonnostaan tapahtuvia virheitä, sekä hyökkäyksiä vastaan. Tässä luvussa esitetään eri tieteellisten tutkimusten johtopäätöksiä parhaista puolustautumisstrategioista koneoppimismallien heikkouksia vastaan.

Terziyanin ym. (2020) arvelevat, että hyökkäykset ovat jo huomattavasti monimutkaisempia kuin mitä tutkimuksissa käsitellään. Suositeltava tapa lähteä parantamaan oppimismallien turvallisuutta, olisikin tutkia ongelman ydintä: tunnistaa kaikki mallien heikkoudet, ymmärtää järjestelmistä johtuvat negatiiviset seuraamukset, sekä kartoittaa hyökkäyksien realistinen todennäköisyys ja niistä johtuva vahinko. Kattavan tutkimuksen tarkoituksena olisi tulevaisuudessa muuttaa oppimismallien toimintaa luotettavammaksi, sekä opettaa ne puolustautumaan automaattisesti hyökkäyksiltä.

Yksi suurimmista ongelmista tekoälyn turvallisuudessa on sen oppimismallien automaattisesta toiminnasta johtuva läpinäkymättömyys. Oppimiseen perustuvia hyökkäyksiä, sekä päätöksenteon epäluotettavuutta on erittäin vaikea karsia, jos malli on rakennettu ottamatta niitä huomioon. Chen ym. (2023) ehdottavat erikseisen koneoppimiseen perustuvan tarkastajamallin lisäämistä tekoälyn rakennus- ja ylläpitoprosessiin. Tarkastajamalli iteratiivisesti tutkisi niin sanotun tarkastuslistan avulla kohdemallin mahdollisia robustisuus ongelmia sen rakennusvaiheen aikana. Havainnon tehdessään, tarkastaja kykenee korjaamaan kohdemallin uudelleenkouluttamalla ongelmaan liittyvän riippuvuuskartoituksen. Ylläpitovaiheessa tarkastajamalli tutkisi kohdemallille annettuja syötteitä adversariaali hyökkäyksien varalta, vertaamalla uusien syötteiden ominaisuuksia tarkastajan vihamielisiksi luokiteltuun dataan.

Myös Terziyan ym. (2020) kannattavat kyberfyysisien järjestelmien älykkyyden suojaamista erikseen toimivalla puolustusmallilla. Tutkijoiden mukaan mallin avulla voitaisiin rakentaa keinotekoinen immuniteetti mm. myrkytyshyökkäyksiä vastaan. Immuniteettijärjestelmä koulutetaan vahvistusoppimisella, jossa koulutusdatana toimii "puhdas"data, sekä vihamieliset esimerkit. Näin rakennettava malli oppii luokittelemaan vihamieliset syötteet hyökkäyksiksi ja osaa automaattisesti suojautua niiltä. Kaikovan ym. (2020) kertovat, kuinka koulutettavan mallin immuniteettiä voidaan kehittää "rokottamalla", eli injektoimalla koulutusdataan uusia syötteitä. Toimenpide on samankaltainen myrkytyshyökkäyksen kanssa, mutta sen tarkoituksena on tarkentaa mallin valinta-alueita injektoimalla hyvä laatuista dataa.



Kuvio 6. Esimerkki rokotusinjektiosta, perustuu Kaikova ym. (2020)

Kuvaajassa näkyy hypoteettinen ihmisen yli- ja alipainoa mittaava malli, jossa x on paino ja y on pituus. Kuvaajasta näemme, kuinka mallin kouluttaminen voi olla koulutusdatasta riippuen epätarkkaa. Esimerkiksi tässä tapauksessa harmaata aluetta on huomattavasti, jolloin mallin päätöksenteko ei ole luotettavaa. Rokottaminen parantaa mallin robustisuutta pienentämällä haavoittuvuus alueita. Nämä alueet ovat myös hyviä kohteita myrkytyshyökkäyksille, joten rokottaminen ennaltaehkäisee myös myrkytyksen vaikutusta. (Kaikova ym. 2022) On huomioitava, että tässä tapauksessa malli ei luokittele normaalipainoista lainkaan.

Immuniteettijärjestelmän kehittäminen puolustautumaan adversariaali hyökkäyksiltä on haastavaa. Vähäkainun ym. (2020) mukaan puolustautumistapoja on monia riippuen kohdemallin rakenteesta ja itse hyökkäyksestä, jota vastaan halutaan puolustautua. Lähtökohtaisesti mallia pitäisi kouluttaa jo aiemmin mainitulla niin sanotulla adversariaali oppimisella, jossa ohjatun oppimisen koulutusdatassa on tarkoituksella injektoituja vihamielisiä esimerkkejä. Tällä oppimistyylillä malli rakentuu robustisemmaksi, koska se oppii tunnistamaan vihamielistä toimintaa automaattisesti.

Kohinaa sisältävien vihamielisten esimerkkien, kuten manipuloitujen kuvasyötteiden käsittelyyn on kehitetty erilaisia keinoja. Vähäkainu ym. (2020) mainitsevat puolustautumistaktiikaksi kohinaa poistavat autoenkooderit. Autoenkoodereiden tarkoitus on lieventää syötteen sisäisiä häiriöitä, pitäen syötteen ja tuloksen kuitenkin samana. Kohinaa poistavat autoenkooderit ovat neuroverkostoja, jotka pienentävät syötteen koodiksi, korjaavat korruptiota tai kohinaa, ja lopuksi rekonstruoivat alkuperäisen syötteen koodin avulla. Kohinaa poistavat autoenkooderit ovat siis koulutettu tunnistamaan vihamielisiä syötteitä ja rekonstruoimaan ne ilman kohinaa. (Zhang 2018) Tutkimuksissa on huomattu myös, kuinka syötteen kompressointi JPEG:i, tai JPEG2000:i on myös auttanut vihamielisten esimerkkien onnistuneessa luokittelussa. (Vähäkainu, Lehto ja Kariluoto, n.d.)

Mallin varastamiselta puolustautumiseen ei ole vielä kehitetty ratkaisua. Vähäkainun ym. (2020) mukaan varastamista voitaisiin estää esimerkiksi kouluttamalla malli havaitsemaan tietyissä hyökkäyksissä käytetyt kyselyt ja syötteet, tai leimaamalla mallin tekemät ennusteet ja luokittelut mallin omistajalle. Leimaamalla tulokset, omistaja voi todistaa alkuperäisen mallin pohjalta rakennettujen kloonien olevan varastettuja. Tämä ei kuitenkaan estä itse hyökkäystä lainkaan, ja hyökkääjä voikin pitää varastetun mallin tiedot yksityisinä, jolloin

leimasta ei ole hyötyä. Nämä strategiat eivät siis välttämättä toimi kokeneita hyökkääjiä vastaan. Juuti ym. (2019) mukaan eräs tapa puolustautua mallin varastamiselta olisi rajoittaa hyökkääjälle palautettua informaatiota rakentamalla ulkopuolinen puolustusmalli havaitsemaan epäilyttävästi toistuvia syötteitä. Puolustusmallin havaittaessa hyökkäyksen, kohdemalli ohjataan muokkaamaan palautettavia tuloksia hyökkääjän hämäämiseksi. Toinen samankaltainen tapa olisi rakentaa malli keräämään ja tutkimaan toistuvia syötteitä. Jos jokin ennalta määritetty kynnys syötteiden ominaisuuksissa ylitetään, malli tietää hyökkäyksen tapahtuvan ja esimerkiksi estää tulevat kyselyt samalta käyttäjältä. Nämä syötteen tutkimiseen perustuvat strategiat ovat tehokkaita, etenkin black-box mallien puolustamisessa, sillä niiden implementointi ei vaadi tietoa puolustettavan mallin rakenteesta, tai koulutusdatasta.

Vähäkainun ym. (2020) mielestä paras tapa puolustautua mallin inversiota vastaan on hyökkäyksen tapahtuessa korruptoida mallin palauttamia tuloksia. Tutkijat esittävät erään tehokkaimmista puolustautumisstrategioista: Differential Privacy (DP). DP-tekniikan tarkoituksena on estää hyökkääjää vastaanottamasta yksityiseen informaatioon perustuvaa koulutusdataa. DP-algoritmit lisäävät mallin tuloksiin satunnaista häiriötä, jotta hyökkääjä ei saa totuuden mukaista tietoa mallin toiminnasta.

6 Yhteenveto

Tässä tutkielmassa käsiteltiin koneoppimiseen perustuvan tekoälyn haavoittuvuuksia, sekä erilaisia puolustautumisstrategioita niiden lieventämiseksi. Tutkimusaihe on erittäin ajankohtainen, sillä tekoälyn luotettavuus vaatii vielä huomattavia parannuksia, etenkin jos sitä käytetään vastuullisissa ympäristöissä ja tehtävissä. Kehitystyö vaatii keskittymistä myös kyberturvallisuuteen, sillä tekoäly on jatkuvasti houkuttelevampi kohde hyökkääjille sen yleistyessä eri aloilla.

Tutkielmassa käsiteltiin tekoälyn oppimiseen liittyviä luonnollisia heikkouksia, kuten robustisuus ongelmia, ennakkoasennoitumista ja maalaisjärjen puutetta. Kognitiivisten heikkouksien ja epäluotettavuuden lisäksi tutkittiin oppimisdataa ja -algoritmeja korruptoivia hyökkäyksiä, kuten tiedonmyrkytystä ja siihen perustuvia hyökkäysstrategioita. Tutkielmassa esiteltiin myös eri adversariaali hyökkäyksiä, joiden tarkoituksena on manipuloida valmiin mallin toimintaa, tai kerätä tietoa sen rakenteesta vihamielisillä syötteillä. Heikkouksien vaaroja konkretisoidaan esittämällä kolme esimerkkitapausta kriittisistä aloista, joissa tekoäly tulee nopeasti yleistymään. Tutkielman lopussa käsitellään eri tutkimuksissa esitettyjä puolustautumisstrategioita kyseisiä heikkouksia vastaan.

Tutkielmassa huomataan, kuinka tärkeää korkealaatuinen ja tarkoituksenmukainen koulutusdata on mallia rakentaessa. Koulutusdataa on validoitava harhaanjohtavan tiedon varalta ja ennakkoasennoitumista on ehkäistävä painottamalla vähemmistöihin liittyvää pienempimääräistä koulutusdataa. Robustisuusongelmia ja hyökkäyksiä vastaan on ehdotettu monia ominaisuuksiltaan eroavia strategioita, mutta useimmissa tutkimuksissa toistuva teema on ulkopuolisen mallin rakentaminen, joka tarkkailee kohdemallin turvallisuutta. Koneoppimismallien täydellinen luotettavuus ja dynaaminen puolustautuminen vaatisi siis lukuisia erirakenteisia malleja, jotka keskittyvät tiettyjen hyökkäyksien tai robustisuusongelmien estämiseen.

Tämän tutkielman tarkoituksena ei ole lietsoa pelkoa tekoälystä ja sen yleistymisestä lähitulevaisuudesta, vaan korostaa kyberturvallisuuden tärkeyttä kognitiivisen laitteiston toiminnassa. Tutkimusdata tästä aiheesta on vielä suhteellisen vähäistä verrattuna kuinka kriittisissä ympäristöissä tekoälyä käytetään, joten on tärkeää lisätä tietämystä sen heikkouksista ja

keskittää resursseja niiden ratkaisemiseksi.

Lähteet

Ahmad, Tanveer, Hongyu Zhu, Dongdong Zhang, Rasikh Tariq, A. Bassam, Fasee Ullah, Ahmed S AlGhamdi ja Sultan S. Alshamrani. 2022. "Energetics Systems and artificial intelligence: Applications of industry 4.0". *Energy Reports* 8:334–361. ISSN: 2352-4847. https://doi.org/10.1016/j.egyr.2021.11.256. https://www.sciencedirect.com/science/article/pii/S2352484721014037.

Chen, Pin-Yu, ja Payel Das. 2023. "AI Maintenance: A Robustness Perspective". *arXiv preprint arXiv:2301.03052,* https://doi.org/10.48550/arXiv.2301.03052.

Chinnamgari, Sunil Kumar. 2019. *R Machine Learning Projects : Implement Supervised, Unsupervised, and Reinforcement Learning Techniques Using R 3.5.* Packt Publishing. ISBN: 9781789807943. https://books.google.fi/books/about/R_Machine_Learning_Projects.html?id=4dKDDwAAQBAJ&redir_esc=y.

Choi, Yejin. 2022. "The Curious Case of Commonsense Intelligence". *Daedalus* 151 (2): 139–155. ISSN: 0011-5266. https://doi.org/10.1162/daed_a_01906. https://doi.org/10.1162/daed_a_01906.

Dadhich, Abhinav. 2018. *Practical Computer Vision: Extract Insightful Information from Images Using TensorFlow, Keras, and OpenCV.* Birmingham, UNITED KINGDOM: Packt Publishing, Limited. ISBN: 978-1-78829-476-8. https://dl.acm.org/doi/book/10.5555/3217393.

Deng, Li, ja Yang Liu, toimittaneet. 2018. *Deep Learning in Natural Language Processing.* Singapore: Springer Singapore : Imprint: Springer. ISBN: 978-981-10-5209-5. https://doi.org/10.1007/978-981-10-5209-5.

Eggers, Shannon Leigh, ja Char Sample. 2020. *Vulnerabilities in Artificial Intelligence and Machine Learning Applications and Data.* Tekninen raportti. Idaho National Lab.(INL), Idaho Falls, ID (United States). https://inldigitallibrary.inl.gov/sites/sti/sti/Sort_57369.pdf.

Ergen, Mustafa. 2019. “What is artificial intelligence? Technical considerations and future perception”. *Anatolian J. Cardiol* 22 (2): 5–7. https://jag.journalagent.com/anatoljcardiol/pdfs/AJC-79091-REVIEW-ERGEN.pdf.

Eykholt, Kevin, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno ja Dawn Song. 2018. “Robust physical-world attacks on deep learning visual classification”. Teoksessa *Proceedings of the IEEE conference on computer vision and pattern recognition,* 1625–1634. https://openaccess.thecvf.com/content_cvpr_2018/papers/Eykholt_Robust_Physical-World_Attacks_CVPR_2018_paper.pdf.

Fernandes de Mello, Rodrigo, ja Moacir Antonelli Ponti. 2018. *Machine Learning: A Practical Approach on the Statistical Learning Theory.* Springer International Publishing. ISBN: 978-3-319-94988-8 978-3-319-94989-5. https://doi.org/10.1007/978-3-319-94989-5. http://link.springer.com/10.1007/978-3-319-94989-5.

Finlayson, Samuel G., John D. Bowers, Joichi Ito, Jonathan L. Zittrain, Andrew L. Beam ja Isaac S. Kohane. 2019. “Adversarial attacks on medical machine learning”. *Science* 363 (6433): 1287–1289. https://doi.org/10.1126/science.aaw4399. https://www.science.org/doi/abs/10.1126/science.aaw4399.

Hu, Charles, ja Yen-Hung Frank Hu. 2020. “Data Poisoning on Deep Learning Models”. Teoksessa *2020 International Conference on Computational Science and Computational Intelligence (CSCI),* 628–632. https://doi.org/10.1109/CSCI51800.2020.00111.

Juuti, Mika, Sebastian Szyller, Samuel Marchal ja N. Asokan. 2019. “PRADA: Protecting Against DNN Model Stealing Attacks”. Teoksessa *2019 IEEE European Symposium on Security and Privacy (EuroSP),* 512–527. https://doi.org/10.1109/EuroSP.2019.00044.

Kaikova, Olena, Vagan Terziyan, Timo Tiihonen, Mariia Golovianko, Svitlana Gryshko ja Liudmyla Titova. 2022. “Hybrid Threats against Industry 4.0: Adversarial Training of Resilience”. Toimittanut R. Absi ja I. El Abbassi. *E3S Web of Conferences* 353:03004. ISSN: 2267-1242. https://doi.org/10.1051/e3sconf/202235303004. https://www.e3s-conferences.org/10.1051/e3sconf/202235303004.

OpenAI. 2021. *ChatGPT.* https://chatgpt.org/.

Russell, Stuart J., Ming-Wei Chang, Jacob Devlin, Anca Dragan, David Forsyth, Ian Goodfellow, Jitendra M. Malik ym. 2021. *Artificial intelligence: a modern approach.* Fourth edition. Pearson series in artificial intelligence. Harlow: Pearson. ISBN: 978-1-292-40117-1. https://www.pearson.com/en-us/subject-catalog/p/artificial-intelligence-a-modern-approach/P200000003500.

Sharma, Anurag, Amanpreet Kaur ja Amit Semwal. 2022. "Supervised and Unsupervised Prediction Application of Machine Learning". Teoksessa *2022 International Conference on Cyber Resilience (ICCR),* 1–5. https://doi.org/10.1109/ICCR56254.2022.9996063.

Skansi, Sandro. 2018. *Introduction to Deep Learning: From Logical Calculus to Artificial Intelligence.* Undergraduate Topics in Computer Science. Springer International Publishing. ISBN: 978-3-319-73003-5 978-3-319-73004-2. https://doi.org/10.1007/978-3-319-73004-2. http://link.springer.com/10.1007/978-3-319-73004-2.

Terziyan, Vagan, Mariia Golovianko ja Svitlana Gryshko. 2020. "Industry 4.0 Intelligence under Attack : From Cognitive Hack to Data Poisoning". Teoksessa *Cyber Defence in Industry 4.0 Systems and Related Logistics and IT Infrastructures,* 110–125. https://jyx.jyu.fi/handle/123456789/60119.

Wang, Kuan-Chieh, Yan Fu, Ke Li, Ashish Khisti, Richard S. Zemel ja Alireza Makhzani. 2022. "Variational Model Inversion Attacks". *CoRR* abs/2201.10787. eprint: 2201.10787. https://arxiv.org/abs/2201.10787.

Vähäkainu, JP, MJ Lehto ja AJE Kariluoto. n.d. "Adversarial Attack's Impact on Machine Learning Model in Cyber-Physical Systems". *Physical Systems,* https://jyx.jyu.fi/handle/123456789/74117.

Vähäkainu, Petri, Martti Lehto, Noëlle van der Waag-Cowling, Louise Leenen, Informaatioteknologian tiedekunta ja Faculty of Information Technology. 2019. *Artificial intelligence in the cyber security environment.* The proceedings of the ... international conference on cyber warfare and security. Academic Conferences International. https://jyx.jyu.fi/handle/123456789/67298.

Zhang, Yifei. 2018. “A better autoencoder for image: Convolutional autoencoder”. Teoksessa *ICONIP17-DCEC*. http://users.cecs.anu.edu.au/~Tom.Gedeon/conf/ABCs2018/paper/ABCs2018_paper_58.pdf.