



Ervin Mägi

Koneoppimisen käyttö keittiöalan ERP-järjestelmässä

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Tieto- ja viestintätekniikka

Insinöörityö

9.3.2023

Tiivistelmä

Tekijä:	Ervin Mägi
Otsikko:	Koneoppimisen käyttö keittiöalan ERP-järjestelmässä
Sivumäärä:	38 sivua
Aika:	9.3.2023

Tutkinto:	Insinööri (AMK)
Tutkinto-ohjelma:	Tieto- ja viestintätekniikka
Ammatillinen pääaine:	Ohjelmistotuotanto
Ohjaajat:	Lehtori Vesa Ollikainen

Tämän opinnäytetyön aiheena on luoda koneoppimissovellus yritykselle, joka toimii keittiöalan toiminnanohjausjärjestelmien parissa. Tavoitteena on ottaa koneoppiminen käyttöön toiminnanohjausjärjestelmässä, jotta toiminnanohjausjärjestelmän asiakkaat voisivat kehittää liiketoimintaansa vähentämällä ruoan hankintakustannuksia. Opinnäytetyössä keskitytään työskentelemään Tensorflow- ja Keras-ohjelmilla, joita tarvitaan koneoppimisen toteuttamiseen. Opinnäytetyön lopputuloksena on Python-ohjelmointikielellä toimiva sovellus, joka pystyy ennustamaan asiakkaalle ruokien menekkejä datan sekä odotettujen säätilatietojen perusteella. Ruokalista luodaan asiakkaan aiempien ruokalistojen ja sääolosuhteiden perusteella. Opinnäytetyössä kuvataan kaikki koneoppimisen vaiheet. Opinnäytetyön lopussa on esitetty vaihe vaiheelta, miten tavoitteet on toteutettu koneoppimisen avulla.

Avainsanat:	Koneoppiminen, Tensorflow, Keras
-------------	----------------------------------

Abstract

Author: Ervin Mägi
Title: Machine Learning in a Foodservice ERP System
Number of Pages: 38 pages
Date: 9 March 2023

Degree: Bachelor of Engineering
Degree Programme: Information and Communication Technology
Professional Major: Software Engineering
Supervisors: Vesa Ollikainen, Senior Lecturer

The aim of the study was to create a machine learning system for a company dealing with ERP kitchen systems. The goal was to implement machine learning in the ERP system so that the customers of the ERP system could develop their business by reducing the cost of food purchases.

The study was carried out using Tensorflow and Keras, which are necessary to create machine learning.

The outcome is an application running in the Python programming language that can predict food sales based on data and expected weather data. The menu is created based on the client's past menus and weather conditions.

The thesis describes all the steps of machine learning and there is a step-by-step showcase of how the objectives were achieved using machine learning.

Keywords: Machine learning, Tensorflow, Keras

Sisällys

Lyhenteet

1	Johdanto	1
2	Mikä on koneoppiminen?	2
3	Koneoppimismallien rakentaminen	3
3.1	Koneoppimisalgoritmit ja -mallit	3
3.2	Mallin toimintaan vaikuttavat tekijät	4
3.2.1	Tiedon määrä ja laatu	4
3.2.2	Pääparametrit	6
3.2.3	Yli- ja alioppiminen	7
3.2.4	Mallin suoritusindikaattorit	8
3.2.5	Optimointialgoritmi ja oppimisnopeus	9
4	Käytetyt teknologiat ja kehitysympäristö	10
4.1	TensorFlow	10
4.2	Keras	12
4.3	Google Colab	13
5	Päätelmät ja prosessikuvaus	14
5.1	Tietojen esikäsittely koneoppimisessa	14
5.2	Tietojen esikäsittely	15
5.3	Mallin rakentaminen	18
5.4	Mallin koulutus	19
5.5	Tulos	23
5.6	Tulosten arviointi	27
6	Tekoälyn tulevaisuus	28
7	Yhteenveto	29
	Lähteet	32

Lyhenteet

- AI: *Artificial intelligence*. Tekoäly on tietokone tai tietokoneohjelma, joka kykenee tekemään älykkäinä pidettäviä toimintoja.
- ERP: *Enterprise Resource Planning*. Toiminnanohjausjärjestelmä on ohjelmisto, joka auttaa organisaatioita liiketoiminnan raportoinnissa ja auttaa kasvattamaan työn tuottavuutta.
- ML: *Machine learning*. Koneoppiminen on tekoälyn osa-alue, jonka tarkoituksena on saada ohjelmisto toimimaan entistä paremmin pohjatiedon ja mahdollisen käyttäjän toiminnan perusteella.
- MLP: *Multilayer perceptron*. Monikerroksinen perseptroniverkko on eräänlainen syöttöverkko, jossa on useita neuronikerroksia ja joka voi oppia monimutkaisia malleja ja suhteita syötetystä datasta, mikä tekee siitä laajalti käytetyn ja tehokkaan arkkitehtuurin erilaisiin koneoppimistehtäviin.
- MSE: *Mean Squared Error*. Keskimääräinen neliövirhe on yleisesti käytetty mitta kahden datasarjan välisen eron mittaamiseen. Sitä käytetään määrittämään ennustettujen arvojen ja todellisten arvojen neliöpoikkeamien keskiarvo.

1 Johdanto

Koneoppiminen auttaa ihmisiä käsittelemään suuren määrän dataa ja tekemään erittäin tarkkoja ennusteita.

Koneoppimisella on keskeinen rooli Jamixin toiminnanohjausjärjestelmän kehittämisessä. Jamix työskentelee ammattikeittiöiden kanssa ja tarjoaa niille älykkään keittiöjärjestelmän, joka tallentaa reseptejä, suunnittelee ruokalistoja sekä hallinnoi tilauksia ja varastoja. [1.]

Tämän opinnäytetyön tavoitteena on käyttää koneoppimista ruokalajien menekin ennustamiseen. Koneoppimisohjelma ennustaa ruokatarjontaa Ilmatieteen laitoksen säätietojen (ja ruokalajin) avulla.

Tämän opinnäytetyön lopussa on esitetty, miten on luotu ohjelmisto, joka luo nopeasti ruokalista vuodeksi kerrallaan. Tätä varten taulukoiden luomisessa käytettiin satunnaisesti valittuja harjoitus- ja testitietoja.

Aluksi harjoitusdataa käytettiin mallin kouluttamiseen. Tämä tehtiin syöttämällä algoritmille joukko syöttötietoja ja antamalla sen oppia ja säätää verkon painoja ajan myötä. Tavoitteena oli minimoida ennustetun ja todellisen tuotoksen välinen virhe. Kun malli oli koulutettu, mallin suorituskkyä arvioitiin satunnaisesti valitun testidatan avulla.

Tämän jälkeen luotiin ennustetut Helsingin ja Jyväskylän ruokalistataulukoita. Tämä tehtiin käyttämällä koulutettua mallia tuottamaan joukko tuotoksia syöttötietojen perusteella. Tuloksena saadut taulukot tarjosivat käyttäjille valikoiman vaihtoehtoja, joista he saattoivat valita odotettujen sääolosuhteiden perusteella.

Sisältö on tarkoitettu henkilöille, jotka eivät tunne koneoppimista ja jotka haluavat saada siitä yleiskäsityksen, jonka pohjalta he voivat syventyä johonkin tiettyyn osa-alueeseen.

2 Mikä on koneoppiminen?

Koneoppiminen on yksi tekoälyn laji, jonka avulla tietokoneet voivat oppia ja tehdä ennusteita tai päätöksiä ilman, että niitä on erikseen ohjelmoitu. Se on tietojenkäsittelytieteen osa-alue ja tekoälyn haara, joka keskittyy sellaisten algoritmien ja tilastollisten mallien kehittämiseen, joiden avulla tietokone voi oppia ja tehdä ennusteita tai päätöksiä tietojen perusteella. [2.]

Koneoppiminen on ollut käsitteenä olemassa jo vuosikymmeniä, mutta se on saanut viime vuosina merkittävää jalansijaa tiedon lisääntyneen saatavuuden sekä tietokonelaitteistojen ja -ohjelmistojen kehittymisen ansiosta.

Koneoppimisalgoritmit voidaan jakaa karkeasti kolmeen luokkaan, joita ovat vahvistusoppiminen, ohjaamaton oppiminen ja vahvistusoppiminen. [3.]

Valvottu oppiminen on yleisin koneoppimisen muoto. Tässä oppimisessa algoritmille annetaan merkittyä dataa, jossa tulo ja lähtötiedot ovat tiedossa. Algoritmi oppii ennustamaan tuotoksen annetusta syötteestä löytämällä datasta kuvioita ja suhteita. Esimerkkejä valvotusta oppimisesta ovat kuvien luokittelu, puheentunnistus ja luonnollisen kielen käsittely. [4.]

Toisaalta valvomattomassa oppimisessä algoritmi koulutetaan merkitsemättömällä datalla, jossa tulosta ei tunneta. Algoritmin on itse löydettävä datasta kuvioita ja suhteita. Esimerkkejä valvomattomasta oppimisesta ovat klusterianalyysi ja poikkeamien havaitseminen. [5.]

Vahvistava oppiminen on koneoppimisen tyyppi, jossa keskitytään opettamaan agenttia tekemään päätöksiä tekemällä ympäristössä toimia palkkiosignaalin maksimoimiseksi. Tämän tyyppistä oppimista käytetään usein robotiikka- ja pelisovelluksissa. [6.]

Koneellista oppimista käytetään monilla aloilla, kuten rahoituksessa, terveydenhuollossa ja vähittäiskaupassa, päätöksenteon parantamiseen, prosessien automatisointiin ja tehokkuuden lisäämiseen. Tämän alan

opinnäytetyö on jatkuvaa ja kehittyy jatkuvasti, ja uusia kehityssuuntia ja sovelluksia löydetään säännöllisesti. [7.]

Yhteenvedona voidaan todeta, että koneoppiminen on nopeasti kehittyvä ala, jota sovelletaan yhä useammin eri toimialoilla päätöksenteon parantamiseen, prosessien automatisointiin ja tehokkuuden lisäämiseen. Teknologian kehittyessä ja tietojen saatavuuden lisääntyessä koneoppimisen mahdollisuudet kasvavat jatkuvasti, mikä tekee siitä jännittävän ja lupaavan alan tutkijoille ja ammattilaisille. [8.]

Kun teknologia kehittyy ja paranee jatkuvasti, koneoppimisella on ratkaiseva rooli teknologian tulevaisuuden muokkaamisessa ja sen saattamisessa kaikkien ulottuville. Olen innoissani voidessani jatkaa uusien sovellusten kehittämistä ihmisten elämän parantamiseksi. [9.]

3 Koneoppimismallien rakentaminen

3.1 Koneoppimisalgoritmit ja -mallit

Koneoppimisen alalla on useita yleisesti käytettyjä algoritmeja, joihin datatieteilijät ja koneoppimisen ammattilaiset luottavat rakentaessaan malleja ja tehdessään ennusteita [10]. Datatieteilijä on asiantuntija, joka kerää, analysoi ja tulkitsee monimutkaisia tietokokonaisuuksia saadakseen niistä merkityksellisiä oivalluksia ja tehdäkseen liiketoimintapäätöksiä [11].

Yksi yleisimmin käytetyistä algoritmeista on lineaarinen regressio, jota käytetään pääasiassa jatkuvan arvon ennustamiseen syötteiden perusteella. Se toimii etsimällä datapistejoukon läpi parhaiten sopivan viivan, jonka avulla voidaan tehdä ennusteita uusista datapisteistä. [12.]

Toinen suosittu algoritmi on logistinen regressio, jota käytetään yleisesti luokittelutehtävissä, kuten sen määrittämisessä, onko sähköposti roskapostia vai ei. Tämä algoritmi toimii etsimällä datapistejoukon läpi parhaiten sopivan

viivan, mutta sen sijaan, että se antaisi jatkuvan arvon, se antaa todennäköisyyden sille, että datapiste kuuluu tiettyyn luokkaan. [13.]

Kolmas suosittu algoritmi on K-means-klusterointi, joka on ohjaamaton oppimisalgoritmi, jota käytetään samankaltaisten datapisteiden ryhmittelyyn, ja sitä käytetään pääasiassa dimensioiden pienentämiseen ja poikkeamien havaitsemiseen [14].

Päätöspuut on toinen suosittu algoritmi, jota käytetään laajalti luokittelu- ja regressiotehtävissä, se rakentaa puumaisen mallin päätöksistä tiettyjen ominaisuuksien perusteella. Jokainen puun sisäinen solmu edustaa tietyn ominaisuuden arvoon perustuvaa päätöstä, ja jokainen lehtisolmu edustaa ennustetta. [15.]

Neuroverkot on mallinnettu ihmisen aivojen rakenteen ja toiminnan mukaan, ja ne koostuvat toisiinsa kytketyistä "neuroneista", jotka voivat käsitellä ja välittää tietoa. Niitä käytetään yleisesti esimerkiksi kuvan- ja puheentunnistukseen. [16.]

Ensemble-menetelmät, kuten Random Forest ja Gradient Boosting, ovat myös suosittuja algoritmeja, jotka yhdistävät useiden yksittäisten mallien ennusteet lopulliseksi ennusteeksi, mikä on hyödyllistä, kun yritetään parantaa mallien tarkkuutta [17].

Nämä ovat joitakin yleisimmin käytetyistä koneoppimisalgoritmeista, ja jokaisella niistä on omat etunsa ja haittansa, ja paras algoritmi riippuu tehtävästä ja tietokokonaisuudesta, jonka kanssa työskentelet. Siksi on tärkeää kokeilla eri algoritmeja parhaan tuloksen antavan algoritmin löytämiseksi. [18.]

3.2 Mallin toimintaan vaikuttavat tekijät

3.2.1 Tiedon määrä ja laatu

Minkä tahansa mallin toiminta riippuu suuresti sen kouluttamiseen ja testaamiseen käytettävien tietojen määrästä ja laadusta. Tarkkojen ja

luotettavien ennusteiden saamiseksi on ratkaisevan tärkeää, että käytössä on suuri ja monipuolinen tietokokonaisuus, joka edustaa kiinnostuksen kohteena olevaa populaatiota. Toisaalta pieni ja puolueellinen tietokanta voi johtaa yli- ja alioppiminen, mikä johtaa mallin heikkoon suorituskyykyyn. [19.]

Ylioppimisesta on kyse silloin, kun koneoppimismallista tulee liian monimutkainen ja se toimii hyvin harjoitusdatalla mutta huonosti uudella datalla, kun taas alioppimisesta on kyse silloin, kun malli on liian yksinkertainen ja se toimii huonosti sekä harjoitusdatalla että uudella datalla [20].

Yksi tärkeimmistä datan määrään vaikuttavista tekijöistä on otoskoko. Suurempi otoskoko mahdollistaa vankemmat ja yleistettävämmät tulokset, koska se lisää mahdollisuuksia saada selville todellinen populaation vaihtelu. Pieni otoskoko voi sitä vastoin johtaa tilastollisen voiman puutteeseen ja epäluotettaviin estimaatteihin. [21.]

Tietojen määrän lisäksi myös tietojen laatu on ratkaisevan tärkeää mallin toimivuuden kannalta. Siihen sisältyvät sellaiset tekijät kuin tietojen täydellisyys, tarkkuus ja edustavuus. Puuttuvia tai epätarkkoja arvoja sisältävä tietokokonaisuus voi aiheuttaa harhaa ja vääristää tuloksia, kun taas tietokokonaisuus, joka ei edusta perusjoukkoa, voi johtaa mallin huonoon suorituskyykyyn, kun sitä sovelletaan uusiin tietoihin. [22.]

Toinen tärkeä tietojen laatuun liittyvä näkökohta on poikkeavien tai ääriarvojen esiintyminen. Näillä arvoilla voi olla suuri vaikutus mallin suorituskyykyyn, ja niitä on käsiteltävä huolellisesti. Ne voidaan esimerkiksi poistaa tai korvata sopivammilla arvoilla sen varmistamiseksi, että mallin ennusteita ei vääristä pieni määrä epänormaaleja havaintoja. [23.]

Yhteenvetona voidaan todeta, että tietojen määrällä ja laadulla on ratkaiseva merkitys tilastollisen mallin toiminnalle. Suuri ja monipuolinen tietokokonaisuus, jossa on laadukkaita arvoja, on olennaisen tärkeää tarkkojen ja luotettavien ennusteiden saamiseksi. Data-analyytikkojen ja -ammattilaisten on otettava

nämä tekijät huolellisesti huomioon, kun he valitsevat ja valmistelevat tietoja mallia varten, sillä ne voivat vaikuttaa suuresti mallin toimintaan. [24.]

3.2.2 Pääparametrit

Koneoppimismallia rakennettaessa ja koulutettaessa on useita perusparametreja, jotka voivat vaikuttaa suuresti sen kokonaissuorituskykyyn. Näihin parametreihin kuuluvat kerrosten ja neuronien määrä, eräköko ja epookit sekä oppimisnopeus. Optimaalisten tulosten saavuttaminen edellyttää, että ymmärretään, miten nämä parametrit vaikuttavat mallin toimintaan. [25.]

Mallin kerrosten ja neuronien määrä voi vaikuttaa suuresti sen suorituskykyyn. Yleensä useammat kerrokset ja neuronit johtavat monimutkaisempaan malliin, mikä voi johtaa parempaan suorituskykyyn harjoitusaineistossa. On kuitenkin tärkeää huomata, että liian monien kerrosten ja neuronien lisääminen voi johtaa ylisovittamiseen, jolloin malli erikoistuu liikaa harjoitusaineistoon ja suoriutuu huonosti tuntemattomassa aineistossa. Siksi on tärkeää löytää tasapaino ja käyttää mahdollisimman vähän kerroksia ja neuroneita, jotta saavutetaan haluttu tarkkuus. [26.]

Eräköko ja epookit ovat kaksi parametria, jotka ohjaavat mallin koulutusprosessia. Eräköko määrittelee yhdessä iteraatiossa käytettävien harjoitusesimerkkien määrän, kun taas epookit määrittelevät iteraatioiden määrän koko tietokokonaisuuden osalta. Näiden kahden parametrin välinen suhde voi vaikuttaa suuresti mallin suorituskykyyn. Suurempi eräköko voi johtaa nopeampaan harjoitteluun, mutta se voi myös johtaa epätarkempaan malliin. Vastaavasti suurempi epookkien määrä voi johtaa tarkempaan malliin, mutta se voi myös lisätä ylisovituksen riskiä. Siksi on tärkeää löytää oikea tasapaino eräkoon ja epookkien välillä. [27.]

Oppimisnopeus on parametri, joka ohjaa sitä, kuinka nopeasti malli mukautuu uuteen tietoon harjoittelun aikana. Suurempi oppimisnopeus voi johtaa nopeampaan harjoitteluun, mutta se voi myös johtaa epätarkempaan malliin. Toisaalta pienempi oppimisnopeus voi johtaa tarkempaan malliin, mutta se voi

myös lisätä koulutusaikaa. Oikean oppimisnopeuden löytäminen on olennaista optimaalisten tulosten saavuttamiseksi. [28.]

Yhteenvedona voidaan todeta, että koneoppimismallin toimintaan vaikuttavien perusparametrien ymmärtäminen on ratkaisevan tärkeää optimaalisen suorituskyvyn saavuttamiseksi. Näihin parametreihin kuuluvat kerrosten ja neuronien määrä, erä koko ja epookit sekä oppimisnopeus. Oikean tasapainon löytäminen näiden parametrien välillä voi johtaa tarkkaan ja tehokkaaseen malliin. Tietotutkijoina on tärkeää kokeilla ja testata jatkuvasti eri parametreja, jotta voidaan löytää paras yhdistelmä tietylle tietokokonaisuudelle ja ongelmalle. [29.]

3.2.3 Yli- ja alioppiminen

Koneoppimisen alalla kaksi yleistä ongelmaa, joita voi esiintyä, ovat ali- ja ylioppiminen. Nämä ongelmat voivat vaikuttaa suuresti mallin suorituskyvyn, ja optimaalisten tulosten saavuttaminen edellyttää, että ymmärretään, miten ne voidaan estää. [30.]

Alioppiminen tapahtuu, kun koneoppimismalli on liian yksinkertainen, se pysty kuvaamaan tarkasti datassa esiintyviä malleja, mikä johtaa huonoon suorituskyvyn sekä harjoitus- että testidatassa. Tämä johtuu usein liian yksinkertaisen mallin käytöstä, harjoittelua varten käytettävissä olevan datan rajallisesta määrästä tai ominaisuuksien sopimattomasta valinnasta.

Alioppimisen välttämiseksi on tärkeää käyttää monimutkaisempaa mallia, lisätä koulutukseen käytettävän datan määrää ja valita relevantteja piirteitä, jotka kuvaavat datassa olevat mallit tarkasti. [31.]

Ylisovittaminen tapahtuu, kun malli on liian monimutkainen ja erikoistuu liikaa harjoitusdataan, mikä johtaa huonoon suorituskyvyn näkymättömällä datalla. Ylisovittamisen ehkäisemiseksi on tärkeää käyttää yksinkertaisempaa mallia, kuratoida ja esikäsitellä dataa asianmukaisesti, vähentää koulutusdatan määrää ja käyttää regularisointitekniikoita. [32.]

Yhteenvetona voidaan todeta, että ali- ja ylisovittaminen ovat kaksi yleistä ongelmaa, joita voi esiintyä koneoppimismallia rakennettaessa ja koulutettaessa. Molemmat ongelmat voivat vaikuttaa suuresti mallin suorituskkyyn, ja niiden estämisen ymmärtäminen on olennaista optimaalisten tulosten saavuttamiseksi. Alisovittamisen estämiseksi on tärkeää käyttää monimutkaisempaa mallia, lisätä mallin kouluttamiseen käytettävän datan määrää ja valita merkitykselliset ominaisuudet. Tietotutkijoina on tärkeää seurata ja arvioida jatkuvasti mallin suorituskkyä näiden ongelmien tunnistamiseksi ja ehkäisemiseksi. [33.]

3.2.4 Mallin suoritusindikaattorit

Koneoppimismallin suorituskkyyn arvioimiseksi käytetään useita mallin suorituskkyindikaattoreita. Näillä indikaattoreilla mitataan mallin tarkkuutta, täsmällisyyttä, palautusta ja yleistä tehokkuutta. Näiden indikaattoreiden ja niiden käytön ymmärtäminen on olennaista optimaalisten tulosten saavuttamiseksi mallia rakennettaessa ja koulutettaessa. [34.]

Tarkkuus on yleisimmin käytetty suorituskkyindikaattori, ja se mittaa mallin tekemien oikeiden ennusteiden prosenttiosuutta. Se lasketaan jakamalla oikeiden ennusteiden määrä ennusteiden kokonaismäärällä. [35.]

Herkkyys-arvoa käytetään myös mallin suorituskkyyn mittaamiseen. Se mittaa oikeiden positiivisten ennusteiden prosenttiosuutta kaikista todellisista positiivisista tapauksista. Korkea herkkyyspistemäärä osoittaa, että malli ei jätä tekemättä monia todellisia positiivisia ennusteita. [36.]

F1-arvo on tarkkuuden ja palautuksen harmoninen keskiarvo. Sitä käytetään yleisesti silloin, kun luokat ovat epätasapainossa tai kun tarkkuudella ja palautuksella on erilainen merkitys. [37.]

ROC-AUC-pistemäärää käytetään mittaamaan vastaanottimen käyttöominaiskäyrän alapuolella olevaa aluetta. Sitä käytetään yleisesti, kun

luokat ovat epätasapainossa tai kun tarkkuudella ja palautuksella on erilainen merkitys. [38.]

Yhteenvedona voidaan todeta, että on olemassa useita mallin suorituskäytännön indikaattoreita, joita käytetään koneoppimismallin suorituskäytännön arviointiin. Näihin indikaattoreihin kuuluvat tarkkuus, täsmällisyys, palautus, F1-arvo ja ROC-AUC-arvo. Näiden indikaattoreiden käytön ja niiden mittaamisen ymmärtäminen on olennaista optimaalisten tulosten saavuttamiseksi mallia rakennettaessa ja koulutettaessa. On tärkeää huomata, että näitä indikaattoreita tulisi käyttää yhdessä eikä keskittyä vain yhteen indikaattoriin. Tietotutkijoina on tärkeää seurata ja arvioida jatkuvasti mallin suorituskäytännön näiden indikaattoreiden avulla mahdollisten ongelmien tunnistamiseksi ja ehkäisemiseksi. [39.]

3.2.5 Optimointialgoritmi ja oppimisnopeus

Koneoppimisen alalla oikean optimointialgoritmin ja oppimisnopeuden valinta voi vaikuttaa suuresti mallin suorituskäytännön. Optimaalisten tulosten saavuttaminen edellyttää sen ymmärtämistä, miten nämä tekijät vaikuttavat mallin toimintaan. [40.]

Optimointialgoritmeja käytetään mallin optimaalisen parametrijoukon löytämiseen. Käytettävissä on useita erilaisia optimointialgoritmeja, joilla kullakin on omat etunsa ja haittansa. Joitakin suosittuja optimointialgoritmeja ovat muun muassa gradienttilaskeutuminen, stokastinen gradienttilaskeutuminen ja Adam. Gradienttilaskeutuminen on yksi yksinkertaisimmista ja yleisimmin käytetyistä optimointialgoritmeista, mutta se voi olla hidas, kun työskennellään suurten tietokokonaisuuksien kanssa. Stokastinen gradienttilaskeutuminen on gradienttilaskeutumisen parannus, joka on nopeampi ja soveltuu paremmin suurille tietokokonaisuuksille. Adam on toinen optimointialgoritmi, jossa yhdistyvät sekä gradienttilaskeutumisen että stokastisen gradienttilaskeutumisen edut, joten se on erinomainen valinta moniin koneoppimisongelmiin. [41.]

Oppimisnopeus on parametri, joka ohjaa sitä, kuinka nopeasti malli mukautuu uuteen tietoon harjoittelun aikana. Sitä käytetään yhdessä optimointialgoritmin kanssa optimaalisen parametrijoukon löytämiseksi mallille. Suurempi oppimisnopeus voi nopeuttaa koulutusta, mutta se voi myös johtaa epätarkempaan malliin. Toisaalta pienempi oppimisnopeus voi johtaa tarkempaan malliin, mutta myös lisätä koulutusaikaa. Oikean oppimisnopeuden löytäminen on olennaista optimaalisten tulosten saavuttamiseksi. [42.]

Johtopäätöksenä voidaan todeta, että oikean optimointialgoritmin ja oppimisnopeuden valitseminen on ratkaisevan tärkeää optimaalisen tuloksen saavuttamiseksi koneoppimismallia rakennettaessa ja koulutettaessa. Optimointialgoritmia käytetään mallin optimaalisen parametrijoukon löytämiseen, kun taas oppimisnopeus ohjaa sitä, kuinka nopeasti malli mukautuu uuteen tietoon koulutuksen aikana. Optimaalisten tulosten saavuttaminen edellyttää sen ymmärtämistä, miten nämä tekijät vaikuttavat mallin toimintaan ja miten niitä käytetään oikein. Tietotutkijoina on tärkeää kokeilla erilaisia optimointialgoritmeja ja oppimisnopeuksia, jotta löydetään paras yhdistelmä tiettyyn tietokokonaisuuteen ja ongelmaan. [43.]

4 Käytetyt teknologiat ja kehitysympäristö

4.1 TensorFlow

TensorFlow on avoimen lähdekoodin ohjelmistokirjasto koneoppimista varten. Sen on kehittänyt Google Brain -tiimi, ja sitä käytetään monenlaisiin tehtäviin, kuten koneoppimisen ja tekoälyn syviin neuroverkkoihin. [44.]

Kirjasto tarjoaa joustavan arkkitehtuurin ja suuren suorituskyvyn mallien kouluttamiseen ja käyttöön otossa. Se tukee monia eri alustoja, kuten Windowsia, Linuxia ja MacOS:ää sekä mobiililaitteita ja sulautettuja laitteita. [45.]

TensorFlow julkaistiin ensimmäisen kerran vuonna 2015, ja sen jälkeen siitä on tullut yksi käytetyimmistä koneoppimisen kehyksistä. Se on suunniteltu

joustavaksi ja mahdollistaa sekä tutkimus- että tuotantokäytön, ja sitä on käytetty monenlaisissa sovelluksissa, kuten kuvan- ja puheentunnistuksessa, luonnollisen kielen käsittelyssä ja aikasarjojen ennustamisessa. [46.]

Yksi TensorFlow'n tärkeimmistä ominaisuuksista on sen kyky käsitellä laajamittaisia, hajautettuja laskutoimituksia, minkä vuoksi se soveltuu hyvin mallien kouluttamiseen suurilla tietokantajoukoilla. Se tarjoaa myös joukon työkaluja laskutoimitusten visualisointiin ja virheenkorjaukseen, mikä on hyödyllistä monimutkaisten mallien sisäisen toiminnan ymmärtämiseksi. [47.]

TensorFlow sisältää ohjelmointimallin, jossa laskutoimitukset esitetään suunnattuna graafina, jota kutsutaan TensorFlow-graafiksi ja jossa solmut edustavat matemaattisia operaatioita ja reunat niiden välillä kulkevaa dataa. Tämä mahdollistaa tehokkaan suorituksen erilaisilla laitteistoilla, kuten suorittimilla ja grafiikkasuorittimilla. [48.]

TensorFlow'lla on suuri ja aktiivinen yhteisö, jossa monet tekijät ja käyttäjät jakavat resursseja ja tarjoavat tukea. Tämä on johtanut monenlaisten kolmansien osapuolten kirjastojen ja työkalujen kehittämiseen, mukaan lukien TensorFlow Serving -mallien tarjoiluun ja TensorFlow Lite -mallien käyttöönottoon mobiililaitteissa ja sulautetuissa laitteissa. [49.]

Äskettäin julkaistu TensorFlow 2 on merkittävä virstanpylväs kehitykselle, sillä se sisältää monia uusia ominaisuuksia ja parannuksia, jotka helpottavat koneoppimismallien kehittämistä ja käyttöönottoa [50].

Yksi merkittävimmistä muutoksista TensorFlow 2:ssa on sen keskittyminen helppokäyttöisyyteen. Kehys sisältää nyt intuitiivisemmän API:n, mikä helpottaa mallien rakentamista ja kouluttamista. Tästä on hyötyä erityisesti aloittelijoille, jotka saattoivat kokea TensorFlow'n edellisen version hieman pelottavana. [51.]

Toinen TensorFlow 2:n keskeinen ominaisuus on Kerasin, suosituksen korkean tason neuroverkkoliittymän, integrointi. Näin kehittäjät voivat rakentaa monimutkaisia neuroverkkomalleja yksinkertaisella, helppolukuisella koodilla.

Tämä yksinkertaistaa entisestään mallien rakentamisprosessia ja tekee siitä helpommin lähestyttävän kaikenlaisille kehittäjille. [52.]

Lisäksi TensorFlow 2 tuo mukanaan tiiviimmän integraation muiden suosittujen kirjastojen, kuten NumPyn ja Pandasin kanssa. Tämä mahdollistaa saumattoman tietojenkäsittelyn, ominaisuuksien suunnittelun ja mallien käyttöönoton. Tämä helpottaa työskentelyä suurten tietokokonaisuuksien kanssa ja mallien kouluttamista pilvipohjaisilla alustoilla. [53.]

Yhteenvetona voidaan todeta, että TensorFlow 2 on merkittävä edistysaskel koneoppimisen alalla, sillä se tarjoaa kehittäjille tehokkaan mutta helppokäyttöisen kehyksen mallien rakentamiseen ja käyttöönottoon. Parannetun API:n, integroidun Kerasin ja parannetun tuen muille suosituille kirjastoille ansiosta TensorFlow 2:lla on hyvät edellytykset tulla koneoppimisen suosituimmaksi kehykseksi tulevana vuosina. [54.]

4.2 Keras

Keras on Python-kielellä kirjoitettu korkean tason syväoppimisliittymä. Se on suunniteltu tekemään neuroverkkojen rakentamisesta, kouluttamisesta ja arvioinnista mahdollisimman yksinkertaista. [55.]

API on rakennettu muiden suosittujen syväoppimiskirjastojen, kuten TensorFlow-, Theano- ja CNTK-kirjastojen päälle, jolloin käyttäjät voivat valita taustamoottorin [56].

Kerasin avulla käyttäjä voi helposti määritellä, kouluttaa ja arvioida malleja jopa muutamalla rivillä koodia. API:n avulla käyttäjä voi ilmaista neuroverkkoja yksinkertaisella, intuitiivisella ja luettavalla syntaksilla. Lisäksi Keras tukee uusimpia arkkitehtuureja, kuten konvoluutio- ja rekursiivisia neuroverkkoja sekä niiden muunnelmia. [57.]

API:n avulla käyttäjä voi myös helposti tallentaa ja ladata malleja, jolloin työn jakaminen, jatkaminen ja parantaminen on helppoa. Keras tukee myös GPU-

pohjaista harjoittelua, jolloin käyttäjä voi kouluttaa suuria malleja nopeammin. [58.]

Yhteenvetona voidaan todeta, että Keras on tehokas, käyttäjäystävällinen ja helppolukuinen API syväoppimiseen Python-kielellä, jota voidaan käyttää tutkimuksessa ja teollisuudessa. Sen avulla käyttäjä voi helposti luoda, kouluttaa ja arvioida neuroverkkoja, ja se on yhteensopiva erilaisten taustakoneiden kanssa. [59.]

4.3 Google Colab

Google Colaboratory, tai lyhyesti Colab, on ilmainen Jupyter-muistikirjaympäristö, jonka avulla voi kirjoittaa ja suorittaa koodia sekä tarkastella visualisointeja ja muita tuotoksia useilla ohjelmointikielillä, kuten Pythonilla. Se soveltuu erityisen hyvin koneoppimistehtäviin, koska se tarjoaa erilaisia työkaluja ja resursseja, joiden avulla ML-mallien kehittäminen ja testaaminen on helppoa. [60.]

Colab tarjoaa pääsyn tehokkaisiin laskentaresursseihin, kuten GPU:t (Graphics Processing Units) ja TPU:t (Tensor Processing Units), joita voidaan käyttää ML-mallien koulutus- ja päättelyprosessien nopeuttamiseen. Tämä tarkoittaa, että vaikka ei olisi käytössä tehokasta tietokonetta, voi silti kouluttaa ja käyttää monimutkaisia ML-malleja, kunhan on internet-yhteys. [61.]

Lisäksi Colab integroituu muihin pilvipalveluihin, joita voidaan käyttää ML-projektien tietojen tallentamiseen ja hallintaan. Tämä helpottaa suurten tietokokonaisuuksien käsittelyä ja skaalausta. [62.]

Colab tukee myös suosittuja avoimen lähdekoodin ML-kehyksiä, kuten TensorFlowta ja PyTorchia, jotka tarjoavat laajan valikoiman valmiita ML-malleja ja kirjastoja, mikä tehostaa ja helpottaa kehitysprosessia [63].

Toinen Colabin hieno ominaisuus on, että sen avulla voi tehdä yhteistyötä muiden kanssa ML-projekteissa jakamalla muistikirjoja ja kutsumalla muita

muokkaamaan ja ajamaan koodia. Tämä voi olla erittäin hyödyllistä ML-projekteissa, joissa useiden ihmisten on työskenneltävä yhdessä, tai kun halutaan jakaa ja saada palautetta malleista. [64.]

Yhteenvetona voidaan todeta, että Colab on erinomainen alusta koneoppimismallien kehittämiseen ja työstämiseen. Se tarjoaa valikoiman tehokkaita laskentaresursseja ja integraation pilvipalveluihin, ja se tukee suosittuja avoimen lähdekoodin koneoppimiskehyksiä, mikä helpottaa koneoppimisprojektien kehittämistä, testaamista ja yhteistyötä. [65.]

5 Päätelmät ja prosessikuvaus

5.1 Tietojen esikäsittely koneoppimisessa

Tietojen esikäsittely on olennainen osa koneoppimisprosessia. Se on prosessi, jossa tiedot valmistellaan koneoppimisalgoritmeja varten. Tiedon esikäsittelyn tarkoituksena on tuottaa puhdasta, jäsenneltyä ja käyttökelpoista dataa, joka mahdollistaa koneoppimismallien tehokkaan harjoittelun.

Datan esikäsittelyssä data puhdistetaan, muunnetaan ja pelkistetään sopivaan muotoon. Datan puhdistamiseen kuuluu tarpeettoman datan poistaminen, puuttuvien arvojen käsittely. Tiedon muuntaminen tarkoittaa tiedon muodon muuttamista, esimerkiksi tekstin muuttamista numeroiksi.

Puhdas, jäsennelty ja käyttökelpoinen data on avainasemassa, jotta koneoppimisalgoritmit voivat toimia tehokkaasti. Jos dataa ei ole puhdistettu ja muotoiltu asianmukaisesti, koneoppimisalgoritmit voivat tuottaa virheellisiä tuloksia. Tämä voi johtaa koneoppimismallin heikkoon suorituskykyyn, ja joissakin tapauksissa se ei välttämättä ole käyttökelpoinen.

Lisäksi datan esikäsittely voi pienentää tietokokonaisuuden kokoa. Tämä voi olla hyödyllistä silloin, kun tietokokonaisuus on liian suuri ja koneoppimismallin kouluttaminen on mahdotonta käytettävissä olevilla laskentaresursseilla.

Datan esikäsittely on siis tärkeä osa koneoppimisprosessia. Se auttaa tuottamaan puhdasta, jäsennettyä ja käyttökelpoista dataa, jonka avulla koneoppimismalleja voidaan kouluttaa tehokkaasti ja tuottaa tarkkoja tuloksia.

5.2 Tietojen esikäsittely

Saadakseni testiruokalistatiedot asiakkaalta yhdistin Jamix Cloudiin ja sain kaikki ruokalistatiedot JSON-muodossa, jota käytin ruokalistan lukemiseen ja syötyjen aterioiden lukumäärän laskemiseen (kuva 2).

Käytin Jyväskylän kaupungin säätietoja, jotka sain Ilmatieteen laitokselta XML-muodossa, mutta tämä projekti toimii samalla tavalla kaikkien muiden Suomen kaupunkien kanssa. Viime vuoden säätietojen perusteella muodostetaan samanlaiset säätiedot seuraavaa vuotta varten. 2021.12.27 on päivämäärä, josta viime vuoden ruokalista alkaa ja johon lisään 334 päivää, jolloin sää mitattiin, ja jonka perusteella luon seuraavan vuoden ruokalistataulukon (kuva 1).

```
city = "Jyvaskyla"  
year_days = 334
```

Kuva 1. Kaupunki, josta säätiedot on otettu opinnäytetyön, ja tutkittujen päivien lukumäärä.

Koulutusdatassa jokainen viikko koostuu viidestä työpäivästä.

```

menus
{
  'postProductionAdditionalInfo': '',
  'foodRanOut': False,
  'foodRanOutTime': None,
  'estimateForAdditionalSales': 0}},
{'menuRowId': 15102,
 'menuRowName': 'Kasvispasta',
 'menuRowPlannedPortions': 50,
 'menuRowPortionWeight': 250.0,
 'menuRowPortionWeightUnit': 'g',
 'menuRowPostProductionData': {'preparedPortions': 50,
 'preparedTotalWeight': 12.5,
 'preparedTotalWeightUnit': 'kg',
 'leftoverPortions': 19,
 'leftoverTotalWeight': 4.75,
 'leftoverTotalWeightUnit': 'kg',
 'salesPortions': 31,
 'salesTotalWeight': 7.75,
 'salesTotalWeightUnit': 'kg',
 'postProductionAdditionalInfo': '',
 'foodRanOut': False,
 'foodRanOutTime': None,
 'estimateForAdditionalSales': 0}},
{'menuRowId': 15158,
 'menuRowName': 'Toscakakku',
 'menuRowPlannedPortions': 50,
 'menuRowPortionWeight': 2251.0,
 'menuRowPortionWeightUnit': 'g',
 'menuRowPostProductionData': {'preparedPortions': 50,
 'preparedTotalWeight': 112.55,
 'preparedTotalWeightUnit': 'kg',
 'leftoverPortions': 19,
 'leftoverTotalWeight': 42.769,
 'leftoverTotalWeightUnit': 'kg',
 'salesPortions': 31,
 'salesTotalWeight': 69.781,
 'salesTotalWeightUnit': 'kg',
 'postProductionAdditionalInfo': ''},

```

Kuva 2. Asiakkaan antama ruokalista JSON-muodossa.

Yhdistin harjoitusaineistoon syötyjen aterioiden määrän per kilo ja ruokalajin nimen (kuva 3). 960 riviä on saatu, kun otetaan huomioon, että 240 päivän ajan on neljä ruokalajia päivässä. Jos jokainen ateria kirjattaisiin erilliselle riville taulukkolaskentaohjelmaan tai tietokokonaisuuteen, rivien kokonaismäärä olisi $240 \times 4 = 960$.

	Portion	Weight (kg)
0	Keitetty riisi	37.0
1	Italian possupata	61.0
2	Kaalikeitto	30.0
3	Toscakakku	33.0
4	Uunibroiler	92.0
...
955	Uunibroiler	83.0
956	Kalakeitto	70.0
957	Italian possupata	57.0
958	Kasvispasta	95.0
959	Keitetty riisi	66.0

960 rows × 2 columns

Kuva 3. Ruokalista taulukossa.

Tämän jälkeen taulukkoon on lisätty Ilmatieteen laitoksen säätiedot (kuva 4).

	Portion	Temperature	Wind Speed	Weight (kg)
0	Keitetty riisi	-10.8	1.9	37.0
1	Italian possupata	-10.8	1.9	61.0
2	Kaalikeitto	-10.8	1.9	30.0
3	Toscakakku	-10.8	1.9	33.0
4	Uunibroiler	-10.3	2.0	92.0
...
955	Uunibroiler	2.6	3.2	83.0
956	Kalakeitto	4.7	2.1	70.0
957	Italian possupata	4.7	2.1	57.0
958	Kasvispasta	4.7	2.1	95.0
959	Keitetty riisi	4.7	2.1	66.0

960 rows × 4 columns

Kuva 4. Yhdistetyt lämpötila-, tuulennopeus- ja ruokalistataulukot.

5.3 Mallin rakentaminen

Kyseinen neuroverkko on monikerroksinen perseptroniverkko (MLP). Syöttö- ja lähtökerrosten välissä oleva piilokerros auttaa verkkoa oppimaan monimutkaisia kuvioita ja suhteita syötetiedoissa. MLP on laajalti käytetty ja tehokas neuroverkkotyyppi erilaisiin koneoppimistehtäviin, ja tämän MLP:n erityinen arkkitehtuuri on huolellisen pohdiskelun ja kokeilujen tulos.

Mallin perusteella ennustetaan ensi vuoden ruokalista. Jotta malli voisi oppia tiedot, jotka se saa käsiteltäväksi, mallin luomiseksi tarvitaan input ja output.

Tieto lautasen nimestä numeron muodossa, lämpötilasta, tuulen nopeudesta lisätään syöttötietoihin (kuva 5). Olen lisännyt tietoja ruokalistojen annospainoista tulosteeseen (kuva 6).

```
array([[ 7. , -10.8,  1.9],
       [ 3. , -10.8,  1.9],
       [ 4. , -10.8,  1.9],
       ...,
       [ 3. ,  4.7,  2.1],
       [ 6. ,  4.7,  2.1],
       [ 7. ,  4.7,  2.1]])
```

Kuva 5. Mallin rakentamiseen tarvittavat lähtötiedot.

```
array([37., 61., 30., 33., 92., 41., 46., 45., 55., 86., 88., 46., 51.,
       31., 41., 97., 76., 91., 67., 75., 48., 33., 98., 59., 79., 57.,
       58., 96., 70., 50., 66., 34., 59., 36., 80., 63., 94., 31., 45.,
       87., 83., 39., 56., 81., 47., 52., 65., 39., 82., 69., 54., 73.,
       96., 53., 86., 43., 48., 90., 37., 78., 42., 97., 55., 32., 68.,
       87., 73., 39., 85., 64., 49., 74., 49., 96., 37., 40., 88., 54.,
       31., 36., 78., 37., 79., 31., 32., 66., 86., 83., 46., 71., 78.,
       58., 51., 36., 74., 78., 65., 33., 93., 56., 99., 57., 44., 72.,
       50., 94., 56., 92., 49., 67., 41., 49., 88., 73., 65., 71., 45.,
       95., 81., 79., 35., 36., 58., 43., 81., 34., 96., 94., 45., 36.,
       85., 74., 73., 97., 83., 41., 43., 39., 34., 62., 58., 30., 97.,
       61., 53., 47., 88., 64., 34., 45., 88., 92., 91., 81., 75., 52.,
       93., 54., 78., 31., 87., 58., 65., 48., 34., 54., 38., 96., 56.,
       39., 65., 31., 85., 66., 93., 98., 67., 54., 95., 41., 43., 46.,
       49., 80., 41., 96., 96., 37., 67., 53., 60., 35., 60., 47., 57.,
       96., 48., 39., 95., 70., 53., 56., 55., 69., 36., 35., 41., 64.,
       41., 33., 67., 92., 62., 85., 45., 64., 84., 38., 90., 32., 96.,
       82., 91., 69., 73., 37., 65., 81., 30., 92., 46., 54., 84., 65.,
```

Kuva 6. Neuroverkon ulostulo on yksi arvo, joka edustaa ennustettua muuttuvaa tavoitemäärää kilogrammoina annetuille syötetiedoille.

5.4 Mallin koulutus

Tässä MLP:ssä ensimmäisessä kerroksessa on 16 neuronia, joissa on ReLU-aktivointifunktio (kuva 7). ReLU-aktivointifunktiota käytetään yleisesti syväoppimisessa, koska se on laskennallisesti tehokas ja sen on osoitettu tuottavan parempia tuloksia kuin muiden aktivointifunktioiden, kuten sigmoidin ja tanhin, avulla. Ensimmäisen kerroksen tulomuoto määritetään tulodatan piirteiden lukumäärällä, jota edustaa `x_features`.

Toinen kerros on yksi neuroni, joka edustaa verkon ulostuloa. Tämä sopii regressio-ongelmaan, jossa tavoitteena on ennustaa jatkuva arvo eikä luokitusmerkki.

Verkko on koottu Adam-optimojilla, joka on suosittu optimointialgoritmi syväoppimisessa, koska se pystyy mukautuvasti säätämään oppimismen nopeutta harjoittelun aikana. Käytetty häviöfunktio on keskimääräinen neliövirhe, joka on yleisesti käytetty häviöfunktio regressio-ongelmissa.

```
modeli = keras.Sequential()

modeli.add(keras.layers.Dense(16, activation='relu', input_shape=(x_features.shape[1],)))
modeli.add(keras.layers.Dense(1))

modeli.compile(optimizer='adam', loss='mean_squared_error')
```

Kuva 7. Malli.

Vuoden ruokalistatiedot koostuvat 240 päivästä, joista kaksi kolmasosaa on jaettu harjoitustietoihin (160 päivää) ja yksi kolmasosa testitietoihin (80 päivää) (kuva 8). Harjoitusjoukko käytetään mallin kouluttamiseen (kuva 9), ja testijoukon avulla arvioidaan mallin suorituskkyä (kuva 10).

```
total_rows = array.shape[0]
training_rows = int(total_rows * (2/3))
training_indices = random.sample(range(total_rows), training_rows)
test_indices = [i for i in range(total_rows) if i not in training_indices]
training_data = array[training_indices,:]
test_data = array[test_indices,:]
```

Kuva 8. Koulutus- ja testitiedot.

	Portion	Temperature	Wind Speed	Weight (kg)
114	11.0	5.8	3.6	65.0
868	15.0	17.9	2.3	34.0
958	6.0	4.7	2.1	95.0
722	4.0	-10.8	1.9	30.0
632	14.0	16.2	2.2	91.0
...
938	4.0	8.8	5.2	90.0
519	7.0	-1.7	2.4	87.0
850	5.0	15.0	4.6	85.0
699	6.0	8.8	5.2	32.0
226	12.0	4.1	2.5	65.0

640 rows × 4 columns

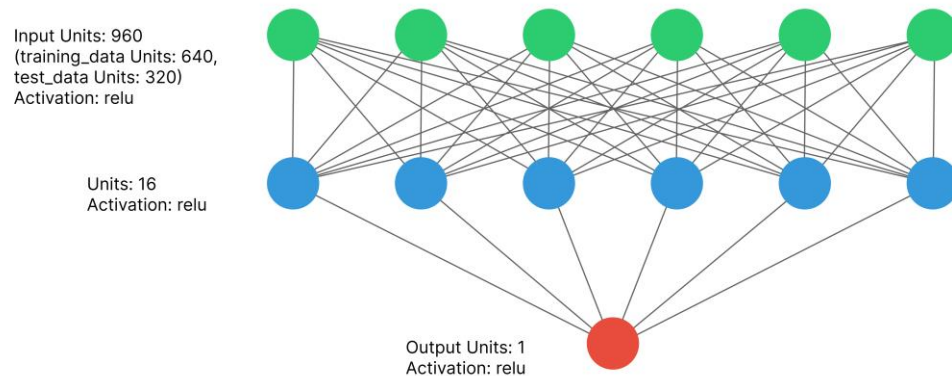
Kuva 9. Satunnaisesti valitut harjoitustiedot.

	Portion	Temperature	Wind Speed	Weight (kg)
148	15.0	17.9	2.3	34.0
277	17.0	-1.7	2.4	31.0
671	1.0	5.7	2.6	35.0
411	1.0	12.6	1.7	31.0
685	12.0	5.3	3.2	35.0
...
815	4.0	2.9	0.8	78.0
949	6.0	1.8	2.7	92.0
427	3.0	7.4	4.1	37.0
531	14.0	-17.7	1.2	73.0
120	1.0	14.2	3.6	35.0

320 rows × 4 columns

Kuva 10. Satunnaisesti valitut testitiedot.

Malli on kuvassa, jossa syöttökerros on vihreällä, piilokerros sinisellä ja tulokerros punaisella (kuva 11).



Kuva 11. Kuvasta näkyy mallin rakenne.

Neuroverkossa piilokerros käsittelee syötteet ja tuottaa tuloksen. Ennustetun tuloksen ja todellisen tuloksen välinen virhe lasketaan MSE:n avulla (kuva 12). Virhettä käytetään sitten mallin painojen päivittämiseen tappion minimoimiseksi. MSE on laajalti käytetty häviöfunktio, koska se on helppo laskea, se on eroteltavissa ja soveltuu monenlaisiin sovelluksiin. [66.]

Lisäksi MSE:tä suositetaan myös siksi, että se rankaisee suurista virheistä enemmän kuin pienistä virheistä, minkä vuoksi se soveltuu malleihin, joissa tarkkuus on tärkeää. Tämä on erityisen tärkeää, kun kyseessä on Keras-malli, jossa on yksi piilokerros ja 960 syötettä, sillä syötteiden suuri määrä tarkoittaa, että pienilläkin ennustusvirheillä voi olla merkittävä vaikutus mallin kokonaistarkkuuteen. [67.]

Yhteenvedona voidaan todeta, että MSE:n käyttö keras-mallissa, jossa on yksi piilokerros ja 960 syötettä, tarjoaa yksinkertaisen ja tehokkaan tavan arvioida mallin suorituskkyä ja auttaa optimoimaan mallin painotuksia. Suurten virheiden rankaiseminen MSE:ssä tekee siitä ihanteellisen sovelluksiin, joissa tarkkuus on merkittävä osatekijä. [68.]

Mean Squared Error: 7.844664411400006

Kuva 12. Mallin keskimääräinen neliövirhe.

5.5 Tulos

Malli on koulutettu sekä harjoitus- (kuva 13) että testidatalla. Luomani malli luo 17 erilaista ruokavaihtoehtoa kunkin päivän koko ruokalistasta. Valitaan 4 parasta vaihtoehtoa, jotka tuovat enemmän voittoa ja asiakkaita. Taulukossa esitetään annospainotietojen ja testitietojen vertailu 240 päivän ajalta, jolloin annospainotiedot sisältävät mittauksia 17 eri elintarvikelajikkeesta. Koska taulukossa verrataan annospainotietoja ja testitietoja kunkin ruokalajikkeen osalta 240 päivän aikana, taulukossa on $240 \times 17 = 4080$ riviä (kuva 14). Harjoitus- ja testidatan välinen ero oli pieni, mikä tarkoittaa, että malli toimii hyvin.

Mallin ennustamien parhaiden tulosten löytämiseksi valittiin automaattisesti ruokalajeja, joissa oli enemmän ruokakiloja, jolloin teoriassa enemmän asiakkaita tulisi syömään, kun he näkisivät ruokalistalla juuri heille sopivaa ruokaa kyseisenä päivänä. Lisäsin päivämäärän tietoihin yhden vuoden, jotta käyttäjä ymmärtää, että tiedot on ennustettu kaikkialle ensi vuodelle.

	Date	Portion	Train set(kg)
0	2022-12-26	Viinimarja-rahkatorttu	33.528873
1	2022-12-26	Vihersalaatti	55.130154
2	2022-12-26	Uunibroiler	28.196026
3	2022-12-26	Toscakakku	34.049007
4	2022-12-26	Suklaatorttu	93.398979
...
4075	2023-11-03	Kalakeitto	82.793953
4076	2023-11-03	Kaalikeitto	66.324516
4077	2023-11-03	Italian possupata	59.359005
4078	2023-11-03	Broiler-kasvislasagne	92.610870
4079	2023-11-03	Aprikoosikiisseli	62.976273

4080 rows × 3 columns

Kuva 13. Harjoitustiedot.

	Date	Portion	Weight (kg)	Test set(kg)	Difference
0	2022-12-26	Viinimarja-rahkatorttu	37.0	32.416672	4.583328
1	2022-12-26	Vihersalaatti	61.0	51.022202	9.977798
2	2022-12-26	Uunibroiler	30.0	39.789398	9.789398
3	2022-12-26	Toscakakku	33.0	29.031830	3.968170
4	2022-12-26	Suklaatorttu	92.0	100.427902	8.427902
...
4075	2023-11-03	Kalakeitto	83.0	78.983429	4.016571
4076	2023-11-03	Kaalikeitto	70.0	72.686119	2.686119
4077	2023-11-03	Italian possupata	57.0	68.606552	11.606552
4078	2023-11-03	Broiler-kasvislasagne	95.0	97.653358	2.653358
4079	2023-11-03	Aprikoosikiisseli	66.0	70.805145	4.805145

4080 rows × 5 columns

Kuva 14. 17 erilaista valikkovaihtoehtoa jokaiselle päivälle, annospainotiedot ja testitiedot sekä niiden välinen ero.

Tuloksena on taulukko, jossa on valittu 4 sopivaa ruokalajia jokaiselle päivälle, joka voidaan näyttää asiakkaille (kuva 15).

	Date	Portion	Weight (kg)
0	2022-12-26	Suklaatorttu	100.427902
1	2022-12-26	Vihersalaatti	72.649208
2	2022-12-26	Uunibroiler	70.161255
3	2022-12-26	Toscakakku	67.784119
4	2022-12-27	Viinimarja-rahkatorttu	74.685089
...
955	2023-11-02	Toscakakku	54.430401
956	2023-11-03	Broiler-kasvislasagne	97.653358
957	2023-11-03	Kalakeitto	78.983429
958	2023-11-03	Kaalikeitto	72.686119
959	2023-11-03	Aprikoosikiisseli	70.805145

960 rows × 3 columns

Kuva 15. Valmis taulukko. Taulukko ennustettu Jyväskylän sään perusteella.

Koulutetun mallin tietoja on käytetty säätietoihin perustuvien ennusteiden toimivuuden tarkistamiseen (kuva 16) ja luotiin Helsingin menuennustetaulukko vuodelle (kuva 17).

	Date	Temperature	Wind Speed
0	2022-12-26	-12.8	1.4
1	2022-12-26	-12.8	1.4
2	2022-12-26	-12.8	1.4
3	2022-12-26	-12.8	1.4
4	2022-12-25	-9.8	1.6
...
955	2023-11-02	-4.4	4.0
956	2023-11-03	0.7	5.0
957	2023-11-03	0.7	5.0
958	2023-11-03	0.7	5.0
959	2023-11-03	0.7	5.0

960 rows × 3 columns

Kuva 16. Helsingin sää.

	Date	Portion	Weight (kg)
0	2022-12-26	Kalakeitto	90.506508
1	2022-12-26	Broiler-kasvislasagne	89.006889
2	2022-12-26	Kasvispasta	88.530830
3	2022-12-26	Italian possupata	86.070442
4	2022-12-27	Italian possupata	92.838280
...
955	2023-11-02	Italian possupata	82.714149
956	2023-11-03	Sekähedelmäkeitto	93.942513
957	2023-11-03	Broiler-kasvislasagne	89.482697
958	2023-11-03	Riistakiusaus	88.539192
959	2023-11-03	Suklaatorttu	82.897606

960 rows × 3 columns

Kuva 17. Taulukko ennustettu Helsingin sään perusteella.

5.6 Tulosten arviointi

Koneoppimismallia rakennettaessa tavoitteena on luoda malli, joka pystyy yleistämään hyvin tuntemattomaan dataan. Tämän vuoksi jaamme datan yleensä harjoitus- ja testijoukkoihin. Malli koulutetaan koulutusjoukolla ja arvioidaan sitten testijoukolla, jotta nähdään, miten hyvin se toimii datassa, jota se ei ole nähnyt koulutuksen aikana.

On tavallista, että ennustussuorituskyky koulutusjoukossa on parempi kuin testijoukossa. Tämä johtuu siitä, että malli on nähnyt harjoitusdatan ja oppinut siinä olevat mallit, mikä johtaa suurempaan tarkkuuteen harjoitusjoukossa. Toisaalta malli ei ole nähnyt testijoukkoa, ja se mittaa mallin kykyä yleistää uusiin, näkemättömiin tietoihin.

Pieni ero koulutus- ja testituloksen välillä on hyvä merkki siitä, että malli ei ole sopeutunut liikaa koulutusdataan. Ylisovittaminen tapahtuu, kun malli on liian monimutkainen ja oppii harjoitusdatan kohinan, mikä aiheuttaa huonon suorituskyvyn testijoukossa. Malli, joka ei ole sovittunut liikaa, on samankaltainen sekä harjoitus- että testijoukoissa.

Jos harjoittelu- ja testisuorituskyvyn välinen ero on kuitenkin pieni, se ei takaa, että malli on täydellinen. Malli voi silti olla tehoton, mikä tarkoittaa, että se on liian yksinkertainen eikä se kuvaa ongelman monimutkaisuutta. Tällöin mallin tarkkuus olisi alhainen sekä harjoitus- että testijoukoissa.

Yhteenvedona voidaan todeta, että harjoittelu- ja testituloksen välinen ero on tärkeä mittari mallin kyvylle yleistää uusiin tietoihin. Kun arvioidaan sääennusteisiin perustuvan mallin suorituskykyä, on myös tärkeää arvioida sen kykyä yleistää se uusiin tietoihin. Tämä tarkoittaa mallin testaamista tiedoilla, joita se ei ole nähnyt koulutuksen aikana, jotta nähdään, pystyykö se tekemään tarkkoja ennusteita. Pieni ero harjoittelu- ja testituloksen välillä on hyvä merkki,

sillä se viittaa siihen, että malli ei sovi liikaa harjoitteluaineistoon ja että se pystyy yleistämään uuteen aineistoon.

6 Tekoälyn tulevaisuus

Tekoälyllä on mahdollisuus mullistaa ravintola-ala parantamalla tehokkuutta, optimoimalla asiakaskokemusta ja lisäämällä myyntiä. Tekoälyn käyttö ravintolaympäristössä todennäköisesti yleistyy tulevaisuudessa entisestään, kun teknologia kehittyy ja yksilöllisten kokemusten kysyntä kasvaa. [69.]

Yksi tärkeimmistä tavoista, joilla tekoälyä voidaan hyödyntää ravintola-alalla, on markkinointi ja asiakassuhteet. Tekoälyjärjestelmät voivat kerätä asiakastietoja, kuten ostohistoriaa, mieltymyksiä ja arvosteluja, ja käyttää niitä henkilökohtaisten markkinointikampanjoiden luomiseen, jotka on räätälöity kunkin asiakkaan tarpeiden ja mieltymysten mukaan. Tekoälyalgoritmit voivat esimerkiksi analysoida asiakkaiden demografisia tietoja ja ostotottumuksia määrittääkseen parhaan ajankohdan ja kanavan kohdistaa erityisiä kampanjoita ja mainontaa kullekin asiakkaalle. Tämä voi lisätä asiakkaiden sitoutumista, uusintamyyntiä ja kokonaismyyntiä. [70.]

Asiakassuhteiden parantamisen lisäksi tekoäly voi myös parantaa ravintoloiden tehokkuutta. Tekoälyalgoritmit voivat esimerkiksi analysoida odotusaikoja, asiakkaiden kysyntää ja henkilöstömäärää optimoidakseen henkilöstömäärän, lyhentääkseen odotusaikoja ja parantaakseen asiakaskokemusta. Tekoäly voi myös automatisoida tehtäviä, kuten varastonhallintaa, jolloin henkilökunnan aikaa vapautuu tärkeämpiin tehtäviin. [71.]

Toinen alue, jolla tekoälyä voidaan käyttää ravintola-alalla, on chatbottien ja virtuaaliavustajien kehittäminen. Nämä tekoälyn avulla toimivat työkalut voivat auttaa asiakkaita tilausten tekemisessä, varausten tekemisessä ja suositusten antamisessa. Ne voivat myös tarjota asiakkaille yksilöllisiä kokemuksia käyttämällä tietoja, kuten asiakkaan mieltymyksiä ja tilaushistoriaa. [72.]

Tekoälyn käyttöön ravintola-alalla liittyy kuitenkin myös eettisiä ja sosiaalisia vaikutuksia, jotka on otettava huomioon. Esimerkiksi asiakastietojen kerääminen ja käyttö herättää huolta yksityisyydestä ja turvallisuudesta. [73.]

Yhteenvedona voidaan todeta, että tekoälyn tulevaisuus ravintola-alalla on lupaava, sillä se voi parantaa tehokkuutta, optimoida asiakaskokemusta ja lisätä myyntiä. On kuitenkin tärkeää ottaa huomioon myös sen käytön eettiset ja sosiaaliset vaikutukset ja ryhtyä ennakoihiin toimiin näiden huolenaiheiden ratkaisemiseksi. Tekoälyteknologian kehittyessä edelleen on tärkeää, että ravintola-ala pysyy ajan tasalla alan uusimmasta kehityksestä ja ottaa sen käyttöön vastuullisesti ja eettisesti. [74.]

7 Yhteenveto

Tässä opinnäytetyössä keskityttiin ravintoloiden vuoden ruokalistojen ennustamiseen, myytyjen annosten määrän kasvattamiseen sekä koneoppimisen ja sen komponenttien tutkimiseen.

Tässä opinnäytetyössä tarkasteltiin koneoppimista ja sen osatekijöitä sekä koneoppimisen soveltamista elintarviketeollisuudessa. Koneoppimisjärjestelmien osien, kuten tietojen esikäsittelyn ja mallin arvioinnin, ymmärtäminen oli tärkeää koneoppimisjärjestelmien tarkkuuden ja tehokkuuden varmistamiseksi.

Ensimmäinen vaihe elintarvikkeen myynnin ennustamismenetelmän toteuttamisessa koneoppimisen avulla oli tietojen kerääminen. Tämä sisälsi tietoja myytävän ruoan tyypistä, ravintolan sijainnista ja sääolosuhteista.

Tensorflowin ja Kerasin avulla valittiin sopiva neuroverkkoarkkitehtuuri ja malli harjoiteltiin esikäsittelyn datan avulla. Tämän jälkeen mallia tarkennettiin ja optimoitiin sen tarkkuuden ja suorituskyvyn parantamiseksi.

Lopuksi mallia voitiin käyttää ennustamaan elintarvikkeiden myyntiä uusien tietojen perusteella.

Kokonaisuudessa elintarvikkeiden myynnin ennustamismenetelmän toteuttaminen koneoppimisessa Tensorflowin ja Kerasin avulla edellytti tietojen keräämistä, neuroverkkomallin kehittämistä, mallin kouluttamista ja optimointia sekä mallin käyttämistä ennusteiden tekemiseen uusien tietojen perusteella.

Koneoppimisen käyttö ravintolateollisuudessa tarjoaa useita jatkokehitysideoita, joita voidaan hyödyntää ruokalistojen ennustamisessa ja myynnin kasvattamisessa entisestään.

Ensinnäkin, koneoppimista voidaan hyödyntää ravintoloiden henkilökunnan hallinnassa ja koulutuksessa. Henkilökunnan käyttäytymistä voidaan seurata ja analysoida koneoppimisen avulla, mikä voi auttaa ravintoloita tunnistamaan tehokkuuden kannalta tärkeitä tekijöitä ja tarjoamaan henkilökunnalle parempaa koulutusta. Lisäksi koneoppimista voidaan käyttää ravintoloiden henkilöstön suunnittelussa, mikä auttaa varmistamaan, että oikea määrä henkilökuntaa on aina käytettävissä ruuhka-aikoina ja vähentää ylityöllistämistä hiljaisina aikoina.

Toiseksi koneoppimista voidaan hyödyntää ruokalajien suunnittelussa.

Koneoppimisalgoritmit voivat analysoida asiakastietoja ja suosituksia, jotta ravintolat voivat kehittää suosittuja ja houkuttelevia ruokalajeja. Tämä voi auttaa ravintoloita erottumaan kilpailijoistaan ja houkuttelemaan enemmän asiakkaita.

Koneoppimisen käyttäminen ruoan kysynnän ennustamiseen ravintola-alalla.

Koneoppimisen avulla luotiin ohjelmisto, joka voi nopeasti luoda vuoden ruokalistan ennustetun ruoan kysynnän perusteella sää- ja ruokalajitietojen avulla. Mallin harjoittelu ja testaus selitetään yksityiskohtaisesti, samoin kuin ruokalistan luominen. Opinnäytetyö on tarkoitettu henkilöille, jotka eivät ole perehtyneet koneoppimiseen ja haluavat saada siitä yleisen käsityksen.

Johtopäätöksessä käsitellään tiedonkeruun merkitystä, sopivan neuroverkkooarkkitehtuurin valintaa ja mallin optimointia, jotta koneoppimismalleissa saavutetaan tarkkuus ja tehokkuus.

Opinnäytetyön tavoitteena oli luoda koneoppimista hyödyntävä ohjelmisto, joka ennustaa ravintoloiden ruoan kysyntää sää- ja lautasinformaation perusteella.

Mallin harjoittelu ja testaus sekä ruokalistojen luominen selvitettiin yksityiskohtaisesti. Opinnäytetyön tavoitteena oli antaa yleinen käsitys koneoppimisesta niille, joille se ei ole tuttu. Johtopäätöksenä voidaan todeta, että kaikki tavoitteet saavutettiin ja ne olivat perusteltuja.

Opinnäytetyössä korostettiin tiedonkeruun tärkeyttä, jota perusteltiin sillä, että tarkat tiedot ovat välttämättömiä, jotta koneoppimisen mallit toimisivat tehokkaasti. Myös sopivan neuroverkkoarkkitehtuurin valintaa käsiteltiin yksityiskohtaisesti ja osoitettiin, miten eri arkkitehtuurit voivat vaikuttaa mallin tarkkuuteen. Myös mallin optimointia tutkittiin, ja sitä perusteltiin sillä, että mallin suorituskkyä voidaan parantaa sen parametreja virittämällä.

Kaiken kaikkiaan tutkielma saavutti onnistuneesti tavoitteensa antamalla kattavan selityksen koneoppimisesta ja osoittamalla sen käytännön sovelluksen ravintola-alalla. Opinnäytetyössä kehitetty ohjelmisto voi auttaa ravintoloita optimoimaan ruokalistojaan ja vähentämään ruokahävikkiä ennustamalla tarkasti ruoan kysynnän. Opinnäytetyö edistää myös koneoppimisen käyttöä elintarviketeollisuudessa käsittelevän kasvavan tutkimuskokonaisuuden täydentämistä.

Lähteet

- 1 Jamix. 2022. Helppoutta ja vastuullisuutta ammattikeittiöön älykkäällä keittiöjärjestelmällä. Verkkoaineisto <<https://www.jamix.com/fi>>. Luettu 29.11.2022.
- 2 Great Learning. 2023. What is Machine Learning? Defination, Types, Applications, and more. Verkkoaineisto <<https://www.mygreatlearning.com/blog/what-is-machine-learning/>>. Luettu 9.2.2023.
- 3 Medium. 2023. What is Supervised Learning in Machine Learning? Verkkoaineisto <<https://arickvigas.medium.com/what-is-supervised-learning-in-machine-learning-bdd98020a5ef>>. Luettu 9.2.2023.
- 4 Guru99. 2023. Supervised Machine Learning: What is, Algorithms with Examples. Verkkoaineisto <<https://www.guru99.com/supervised-machine-learning.html>>. Luettu 9.2.2023.
- 5 Towards data science. 2019. Machine Learning, Part 3: Unsupervised Learning. Verkkoaineisto <<https://towardsdatascience.com/machine-learning-part-3-unsupervised-learning-6d9e59924c34>>. Luettu 29.11.2022.
- 6 Towards data science. 2018. Reinforcement Learning 101. Verkkoaineisto <<https://towardsdatascience.com/reinforcement-learning-101-e24b50e1d292>>. Luettu 29.11.2022.
- 7 CIOinsight. 2022. Is Machine Learning the Same as Predictive Analytics? Verkkoaineisto <<https://www.cioinsight.com/big-data/machine-learning-predictive-analytics/>>. Luettu 29.11.2022.
- 8 Wikipedia. 2022. Machine learning. Verkkoaineisto <https://en.wikipedia.org/wiki/Machine_learning>. Luettu 14.12.2022.
- 9 Google Developers. 2022. Introduction to Machine Learning. Verkkoaineisto <<https://developers.google.com/machine-learning/crash-course/ml-intro>>. Luettu 14.12.2022.
- 10 Rapidminer. 2020. 10 Common Machine Learning Algorithms You Need to Know. Verkkoaineisto <<https://rapidminer.com/blog/10-machine-learning-algorithms/>>. Luettu 14.12.2022.
- 11 Medium. 2022. data scWho is a Data Scientist? What do they do? — Job Description. Verkkoaineisto <<https://datascience->

learnbay.medium.com/who-is-a-data-scientist-what-do-they-do-job-description-43dba07ee45b>. Luettu 14.12.2022.

- 12 Towards data science. 2021. 7 of the Most Used Regression Algorithms and How to Choose the Right One. Verkkoaineisto <<https://towardsdatascience.com/7-of-the-most-commonly-used-regression-algorithms-and-how-to-choose-the-right-one-fc3c8890f9e3>>. Luettu 14.12.2022.
- 13 IBM. 2022. What is logistic regression? Verkkoaineisto <<https://www.ibm.com/topics/logistic-regression>>. Luettu 14.12.2022.
- 14 Towards data science. 2020. 11 Most Common Machine Learning Algorithms Explained in a Nutshell. Verkkoaineisto <<https://towardsdatascience.com/11-most-common-machine-learning-algorithms-explained-in-a-nutshell-cc6e98df93be>>. Luettu 14.12.2022.
- 15 Medium. 2020. Decision Tree. Verkkoaineisto <<https://medium.com/analytics-vidhya/decision-tree-bc70b3f3b110>>. Luettu 14.12.2022.
- 16 Data camp. 2022. 8 Machine Learning Models Explained in 20 Minutes. Verkkoaineisto <<https://www.datacamp.com/blog/machine-learning-models-explained>>. Luettu 14.12.2022.
- 17 Medium. 2021. Ensemble methods and why they are most preferred among Machine Learning practitioners. Verkkoaineisto <<https://medium.com/almabetter/ensembling-methods-and-why-they-are-most-preferred-among-machine-learning-practitioners-e0d6368e994e>>. Luettu 14.12.2022.
- 18 Kaggle. 2022. Common Machine Learning Algorithms. Verkkoaineisto <<https://www.kaggle.com/code/filterjoe/common-machine-learning-algorithms>>. Luettu 14.12.2022.
- 19 Monolithai. 2022. Data Quality and Quantity for Machine Learning. Verkkoaineisto <<https://www.monolithai.com/blog/data-quality-and-quantity-for-machine-learning>>. Luettu 14.12.2022.
- 20 Towards data science. 2020. Understanding Overfitting and Underfitting. Verkkoaineisto <<https://towardsdatascience.com/understanding-overfitting-and-underfitting-12c927e3b9ea>>. Luettu 14.12.2022.
- 21 Towards data science. 2020. The Relationship between Significance, Power, Sample Size & Effect Size. Verkkoaineisto <<https://towardsdatascience.com/the-relationship-between-significance-power-sample-size-effect-size-899fcf95a76d>>. Luettu 14.12.2022.

- 22 Alation. 2021. What Is Data Quality and Why Is It Important? Verkkoaineisto <<https://www.alation.com/blog/what-is-data-quality-why-is-it-important/>>. Luettu 14.12.2022.
- 23 Towards data science. 2020. Data Quality for Everyday Analysis. Verkkoaineisto <<https://towardsdatascience.com/data-quality-for-everyday-analysis-d3aa1442c31>>. Luettu 14.12.2022.
- 24 Data camp. 2022. Data Demystified: Quantitative vs. Qualitative Data. Verkkoaineisto <<https://www.datacamp.com/blog/data-demystified-quantitative-vs-qualitative-data>>. Luettu 14.12.2022.
- 25 Medium. 2018. Train machine learning model — artificial intelligence. Verkkoaineisto <<https://medium.com/idadatoday/4-train-machine-learning-model-artificial-intelligence-e164ff825d1b>>. Luettu 14.12.2022.
- 26 Medium. 2020. Why we need Multi Layer Neural Network (MLP)? Verkkoaineisto <https://medium.com/@Rohit_Varma/why-we-need-multi-layer-neural-network-mlp-d50425b8f37d>. Luettu 14.12.2022.
- 27 Medium. 2021. Difference Between the Batch size and Epoch in Neural Network. Verkkoaineisto <<https://medium.com/mlearning-ai/difference-between-the-batch-size-and-epoch-in-neural-network-2d2cb2a16734>>. Luettu 14.12.2022.
- 28 Towards data science. 2019. Exploring Learning Rates to improve model performance in Keras. Verkkoaineisto <<https://towardsdatascience.com/exploring-learning-rates-to-improve-model-performance-in-keras-e37f5e63f16c>>. Luettu 14.12.2022.
- 29 Towards data science. 2019. Learning Parameters, Part 0: Basic Stuff. Verkkoaineisto <<https://towardsdatascience.com/learning-parameters-part-0-5cfff647bdc>>. Luettu 28.12.2022.
- 30 Towards data science. 2019. Underfitting and Overfitting in machine learning and how to deal with it!!! Verkkoaineisto <<https://towardsdatascience.com/underfitting-and-overfitting-in-machine-learning-and-how-to-deal-with-it-6fe4a8a49dbf>>. Luettu 28.12.2022.
- 31 Wikipedia. 2022. Oversampling and undersampling in data analysis. Verkkoaineisto <https://en.wikipedia.org/wiki/Oversampling_and_undersampling_in_data_analysis>. Luettu 28.12.2022.
- 32 Towards data science. 2022. Overfitting in ML: Understanding and Avoiding the Pitfalls. Verkkoaineisto

- <<https://towardsdatascience.com/overfitting-in-ml-avoiding-the-pitfalls-d5225b7118d>>. Luettu 28.12.2022.
- 33 Kaggle. 2022. Underfitting and Overfitting. Verkkoaineisto <<https://www.kaggle.com/code/dansbecker/underfitting-and-overfitting>>. Luettu 28.12.2022.
 - 34 Medium. 2020. Understanding Performance metrics for Machine Learning Algorithms. Verkkoaineisto <<https://medium.com/analytics-vidhya/understanding-performance-metrics-for-machine-learning-algorithms-996dd7efde1e>>. Luettu 28.12.2022.
 - 35 Hasty. 2022. Documentation. Verkkoaineisto <<https://hasty.ai/docs/mp-wiki/metrics/accuracy>>. Luettu 28.12.2022.
 - 36 Towards data science. 2020. A Look at Precision, Recall, and F1-Score. Verkkoaineisto <<https://towardsdatascience.com/a-look-at-precision-recall-and-f1-score-36b5fd0dd3ec>>. Luettu 28.12.2022.
 - 37 BuiltIn. 2022. Use Precision and Recall to Evaluate Your Classification Model When Accuracy Isn't Enough. Verkkoaineisto <<https://builtin.com/data-science/precision-and-recall>>. Luettu 28.12.2022.
 - 38 Towards data science. 2018. Understanding AUC - ROC Curve. Verkkoaineisto <<https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5>>. Luettu 28.12.2022.
 - 39 Neptune. 2023. F1 Score vs ROC AUC vs Accuracy vs PR AUC. Verkkoaineisto <<https://neptune.ai/blog/f1-score-accuracy-roc-auc-pr-auc>>. Luettu 2.1.2023.
 - 40 Lightly. 2023. Which Optimizer should I use for my ML Project? Verkkoaineisto <<https://www.lightly.ai/post/which-optimizer-should-i-use-for-my-machine-learning-project>>. Luettu 2.1.2023.
 - 41 Programmer Sought. 2023. The most commonly used optimization algorithm-gradient descent method. Verkkoaineisto <<https://www.programmersought.com/article/68165247149/>>. Luettu 2.1.2023.
 - 42 Towards data science. 2018. Understanding Learning Rates and How It Improves Performance in Deep Learning. Verkkoaineisto <<https://towardsdatascience.com/understanding-learning-rates-and-how-it-improves-performance-in-deep-learning-d0d4059c1c10>>. Luettu 2.1.2023.
 - 43 Towards data science. 2018. What lies beneath? Optimization at the heart of Machine Learning. Verkkoaineisto <<https://towardsdatascience.com/a->

- [quick-overview-of-optimization-models-for-machine-learning-and-statistics-38e3a7d13138](#)>. Luettu 4.1.2023.
- 44 Towards AI. 2021. What is TensorFlow, and how does it work? Verkkoaineisto <<https://pub.towardsai.net/what-is-tensorflow-and-how-does-it-work-4d40bba53fd7>>. Luettu 4.1.2023.
 - 45 Pypi. 2022. tensorflow-macos 2.11.0. Verkkoaineisto <<https://pypi.org/project/tensorflow-macos/>>. Luettu 4.1.2023.
 - 46 Medium. 2018. TensorFlow: an Open-Source AI Platform. Verkkoaineisto <https://medium.com/@v_c_q/tensorflow-an-open-source-ai-platform-188bd6a69393>. Luettu 4.1.2023.
 - 47 Medium. 2021. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. Verkkoaineisto <<https://medium.com/mllearning-ai/tensorflow-large-scale-machine-learning-on-heterogeneous-distributed-systems-1c16cd1efacb>>. Luettu 4.1.2023.
 - 48 DZone. 2018. TensorFlow in a Nutshell. Verkkoaineisto <<https://dzone.com/articles/tensorflow-in-a-nutshell>>. Luettu 6.1.2023.
 - 49 Wikipedia. 2023. TensorFlow. Verkkoaineisto <<https://en.wikipedia.org/wiki/TensorFlow>>. Luettu 6.1.2023.
 - 50 Level up. 2019. 5 Important Changes Coming with TensorFlow 2.0. Verkkoaineisto <<https://levelup.gitconnected.com/5-important-changes-coming-with-tensorflow-2-0-e6bb172c5fdf>>. Luettu 6.1.2023.
 - 51 Medium. 2019. Google Debuts TensorFlow 2.0 Alpha. Verkkoaineisto <<https://medium.com/syncedreview/google-debuts-tensorflow-2-0-alpha-ef1e01180c29>>. Luettu 6.1.2023.
 - 52 Towards data science. 2019. TensorFlow is in a relationship with Keras — Introducing TF 2.0. Verkkoaineisto <<https://towardsdatascience.com/tensorflow-is-in-a-relationship-with-keras-introducing-tf-2-0-dcf1228f73ae>>. Luettu 6.1.2023.
 - 53 Towards data science. 2020. Beginner's Guide to TensorFlow 2.x for Deep Learning Applications. Verkkoaineisto <<https://towardsdatascience.com/beginners-guide-to-tensorflow-2-x-for-deep-learning-applications-c7ebd0dcfbee>>. Luettu 6.1.2023.
 - 54 Machine Learning Mastery. 2019. TensorFlow 2 Tutorial: Get Started in Deep Learning with tf.keras. Verkkoaineisto <<https://machinelearningmastery.com/tensorflow-tutorial-deep-learning-with-tf-keras/>>. Luettu 8.1.2023.

- 55 Morioh. 2022. Keras: A Deep Learning API Written In Python. Verkkoaineisto <<https://morioh.com/p/0bfd92b116f>>. Luettu 8.1.2023.
- 56 Guru99. 2023. Keras Tutorial: What is Keras? How to Install in Python [Example]. Verkkoaineisto <<https://www.guru99.com/keras-tutorial.html>>. Luettu 8.1.2023.
- 57 Tutorial and example. 2019. Keras Tutorial. Verkkoaineisto <<https://www.tutorialandexample.com/keras-tutorial>>. Luettu 8.1.2023.
- 58 Keras. 2021. Why choose Keras? Verkkoaineisto <https://keras.io/why_keras/>. Luettu 8.1.2023.
- 59 Wikipedia. 2023. Keras. Verkkoaineisto <<https://en.wikipedia.org/wiki/Keras>>. Luettu 8.1.2023.
- 60 Opendgenus. 2022. Beginner's Guide to Google Colaboratory. Verkkoaineisto <<https://iq.opendgenus.org/google-colab/>>. Luettu 8.1.2023.
- 61 ByteXD. 2022. What is Google Colab: A Beginner's Guide. Verkkoaineisto <<https://bytexd.com/what-is-google-colab-a-beginner-guide/>>. Luettu 8.1.2023.
- 62 Towards data science. 2020. 4 Reasons Why You Should Use Google Colab for Your Next Project. Verkkoaineisto <<https://towardsdatascience.com/4-reasons-why-you-should-use-google-colab-for-your-next-project-b0c4aad39ed>>. Luettu 8.1.2023.
- 63 Tutorialspoint. 2017. Google Colab - Quick Guide. Verkkoaineisto <https://www.tutorialspoint.com/google_colab/google_colab_quick_guide.htm>. Luettu 8.1.2023.
- 64 Medium. 2020. Machine Learning: Google Colab- Why, When and How to Use it. Verkkoaineisto <<https://medium.com/swlh/machine-learning-google-colab-why-when-and-how-to-use-it-9624e34abd6d>>. Luettu 8.1.2023.
- 65 Towards data science. 2022. Google Colab: An Online Jupyter Notebook That You Should Definitely Try. Verkkoaineisto <<https://towardsdatascience.com/google-colab-an-online-jupyter-notebook-that-you-should-definitely-try-2572a3d4afb6>>. Luettu 8.1.2023.
- 66 The AI dream. 2021. Loss Functions in Neural Networks. Verkkoaineisto <<https://www.theaidream.com/post/loss-functions-in-neural-networks>>. Luettu 8.1.2023.

- 67 Stephen Allwright. 2022. What is a good MSE value? (simply explained). Verkkoaineisto <<https://stephenallwright.com/good-mse-value/>>. Luettu 8.1.2023.
- 68 Quora. 2018. Why do we use a mean squared error in a neural network? Verkkoaineisto <<https://www.quora.com/Why-do-we-use-a-mean-squared-error-in-a-neural-network>>. Luettu 12.2.2023.
- 69 Go tenzo. 2022. Artificial intelligence and restaurants: a recipe for success? Verkkoaineisto <<https://blog.gotenzo.com/artificial-intelligence-and-restaurants-a-recipe-for-success>>. Luettu 12.2.2023.
- 70 Deputy. 2019. Benefits of Artificial Intelligence in the Restaurant Industry. Verkkoaineisto <<https://www.deputy.com/blog/benefits-of-artificial-intelligence-in-the-restaurant-industry>>. Luettu 12.2.2023.
- 71 Aglowid. 2020. Artificial Intelligence AI in Restaurant Business – Benefits, Possibilities & Future. Verkkoaineisto <<https://aglowiditsolutions.com/blog/ai-in-restaurant-business/>>. Luettu 12.2.2023.
- 72 Popmenu. 2022. AI in Restaurants: 9 Ways Artificial Intelligence Is Shaping the Food Industry. Verkkoaineisto <<https://get.popmenu.com/post/ai-in-restaurants>>. Luettu 12.2.2023.
- 73 IBM. 2022. AI Ethics. Verkkoaineisto <<https://www.ibm.com/topics/ai-ethics>>. Luettu 12.2.2023.
- 74 Softweb solutions. 2019. AI is proving to be a game-changer for the F&B industry. Verkkoaineisto <<https://www.softwebsolutions.com/resources/AI-in-food-beverage-industry.html>>. Luettu 12.2.2023.