TURUN YLIOPISTO

Matemaattisten mallien ja syväoppimisen käyttö puheentunnistuksessa

Tietojenkäsittelytiede
Tietojenkäsittelytieteiden tutkinto-ohjelma
Tietotekniikan laitos, Teknillinen tiedekunta
Kandidaatin tutkielma

Laatija:
Ilkka Suominen

Lokakuu 2025

Kandidaatin tutkielma
Tietotekniikan laitos, Teknillinen tiedekunta
Turun yliopisto

Oppiaine: Tietojenkäsittelytiede
Tutkinto-ohjelma: Tietojenkäsittelytieteiden tutkinto-ohjelma
Tekijä: Ilkka Suominen
Otsikko: Matemaattisten mallien ja syväoppimisen käyttö puheentunnistuksessa
Sivumäärä: 20 sivua
Päivämäärä: Lokakuu 2025

Puheentunnistus on tekoälyn sovelluskohde, jonka tarkoituksena on muuttaa ihmisen tuottama puhe tietokoneen ymmärtämään muotoon ja tunnistaa siitä kielellisesti merkityksellisiä rakenteita mahdollisimman virheettömästi. Tämän saavuttamiseksi tarvitaan fonetiikan, signaalinkäsittelyn ja koneoppimisen käsitteiden yhdistämistä. Puheentunnistus voidaan toteuttaa tietoteknisesti käyttämällä joko perinteisiä matemaattisia malleja kuten piilotettuja Markovin malleja tai syväoppimismalleja, joiden suosio on kasvanut 2000-luvulta alkaen aina tähän päivään asti. Kummankin tyypin malleissa kielen semanttisten piirteiden esittämiseen käytetään sanavektoreita.

Syväoppimisen merkittävin ero perinteisiin malleihin verrattuna on käytettyjen verkkokerrosten määrässä, joka on syväoppimisverkoissa huomattavasti suurempi kuin perinteisten matemaattisten mallien käyttämissä verkoissa. Puheentunnistuksessa perinteisillä matemaattisilla malleilla ja syväoppimisella on kummallakin omat hyvät puolensa riippuen käyttökohteen monimutkaisuudesta. Syväoppimismallit ovat suuremman verkkokerrosten määrän vuoksi vähemmän riippuvaisia ominaisuuksien luokittelusta, kun taas perinteisillä malleilla niiden suorituskyky perustuu siihen, että ominaisuudet pystytään luokittelemaan tarkasti. Syväoppimismallit ovat täten parempia hyödyntämään ohjaamatointa oppimista. Syväoppimismallien etuna on myös se, että tekoälyn viimeaikainen kehitys on tapahtunut sellaisilla alueilla, jotka ovat läheisesti yhteydessä syväoppimiseen.

Vaikka syväoppimismallit ovat yleisesti ottaen perinteisiä malleja tehokkaampia, joissain tutkimuksissa on todettu, että perinteisten mallien ja syväoppimismallien rakenteita yhdistelevät ns. hybridimallit ovat suorituskyvyltään tietyissä puheentunnistuksen sovelluksissa parempia kuin kumpikaan edellisistä malleista yksin. Hybridimallien vahvuuksiksi mainittiin pienemmän virhemäärän lisäksi myös parempi suorituskyky ja laskennallisella kuormittavuudella mitattuna kevyempi mallinnuskyky.

Asiasanat: puheentunnistus, koneoppiminen, syväoppiminen

Sisällysluettelo

1 Johdanto

Nykyään päivittäisessä arjessa useimmiten puheentunnistuksen kanssa päätyy tekemisiin erilaisten puheenkäsittelysovellusten kautta. Tällaisia ovat esimerkiksi puheen koodaus, ”puheesta tekstiksi”-muunnos, puhujan tunnistus ja varmennus, puheenparannus, kielentunnistus, puheen transkriptio, puhujan asenteiden ja tunteiden tunnistus, audiovisuaalinen signaalinkäsittely ja puhutun dialogin järjestelmät (Kapargavali & Chandra, 2016). Merkittäväksi puheentunnistuksen sovelluskohteeksi ovat aivan viime vuosina nousseet erilaiset virtuaaliavustajat, kuten esimerkiksi Amazonin Alexa, Applen Siri, Microsoftin Cortana ja Googlen Google Assistant. Niiden ominaisuudet poikkeavat hieman toisistaan, mutta ne pystyvät ääniohjauksella muun muassa tekemään internet-hakuja, asettamaan muistutuksia, tekemään ostoksia, lukemaan kirjoja, tekemään pöytävarauksia ravintoloihin ja hallitsemaan sähköpostiohjelmia. (Reis ym., 2017.) Näiden sovellusten taustalla oleva teoria on kuitenkin jo huomattavan vanhaa ja niiden toiminta tämänhetkisessä laajuudessaan ja luotettavuudessaan on vaatinut useiden vuosikymmenten teknologisen kehityksen ja edelleenkin tästä huolimatta puheentunnistuksen käyttötilanteissa sattuu usein väärinymmärryksiä ja virhetilanteita. Käyttäjän näkökulmasta näiden virhetilanteiden ja puheentunnistusjärjestelmien rajoitusten ymmärtäminen vaatii tietoa siitä, kuinka puheentunnistus on teknisesti toteutettu. Matemaattinen mallinnus käyttäen joko perinteisiä matemaattisia malleja tai syväoppimista on keskeinen osa tätä toteutusta.

Puheentunnistuksen prosessi on seuraava: puhe tallennetaan ja muutetaan digitaaliseen muotoon, tämän jälkeen digitoitu puhe pilkotaan pieniin paloihin, joista jokaisesta muodostetaan sen aaltomuodon sisältämän taajuusinformaation muodostama piirrevektori. Näitä piirrevektoreita verrataan tunnettujen äänteiden piirrevektoreihin, joista muodostetaan kyseisen puhenäytteen äännemalli. Tässä äännemallissa olevia virheitä voidaan tunnistaa vertaamalla sitä kyseisen puhekielen kielimalliin, joka arvioi kuinka todennäköisiä tarkasteltavan äännemallin äännerakenteet ovat kohdekielen kontekstissa. (Kurimo, 2009.) 1970-luvulta alkaen tähän päivään asti puheen tietokoneavusteinen käsittelyteknologia on kehittynyt suurin harppauksin. Tähän on akateemisessa kirjallisuudessa viitattu historiallisesti käsitteellä automaattinen puheentunnistus (engl. Automatic Speech Recognition, ASR). ASR-järjestelmä on tehty monista komponenteista, mukaan lukien puhesignaalin esikäsittely, piirteiden erottaminen, akustinen mallinnus, foneettisten yksiköiden tunnistus ja kielen mallinnus. Perinteiset ASR-järjestelmät integroivat sekä piilotetut Markovin mallit (engl.

Hidden Markov Models, HMM) että Gaussin sekoitusmallit (Gaussian Mixture Models, GMM). HMM:a käytetään sellaisen puheen vaihtelun käsittelyyn, joka liittyy aika-avaruuteen, kun taas GMM:t edustavat ääniyksiköiden akustisia ominaisuuksia. Mallinnusprosessi on aikaa vievä ja vaatii erittäin suuren harjoitustietojoukon korkean tarkkuuden saavuttamiseksi. (Pouyanfar ym., 2019.)

Puheentunnistuksen suosiota sovelluskohteena heijastuu myös sen suosioon akateemisen tutkimuksen aiheena, johon on viimeisen noin 15 vuoden aikana liittynyt yhä enemmän syväoppiminen. Esimerkiksi Turun Yliopiston Volter-kirjastotietokantaan vuoden 2024 huhtikuussa tehty haku hakusanoilla ”speech recognition” AND ”deep learning” rajattuna vuoteen 2017 ja sitä uudempiin materiaaleihin tuotti kaikkiaan 5770 tulosta, joista 2600 oli tieteellisiä artikkeleita, 2242 patentteja, 517 konferenssijulkaisuja, 138 kirjan lukua ja 125 väitöskirjaa ja muuta opinnäytettä.

Tämän massiivisen kirjallisuuden määrän vuoksi tämä tutkielma on rajattu antamaan lukijalle yleiskuvaus matemaattisista malleista käyttämällä esimerkkinä niistä yleisintä, HMM-mallia. Syväoppimismalleista on valittu muutama tarkempaan tarkasteluun sen perusteella, kuinka laajasti niitä esitellään aiemmissa syväoppimista puheentunnistuksessa käsittelevissä kirjallisuuskatsauksissa. Tutkielman rakenne on seuraava: luvussa 2 aihetta taustoitetaan kertomalla puheen tuottamisesta ja havaitsemisesta; luvussa 3 kerrotaan kuinka puhe muutetaan digitaaliseen muotoon; luku 4 käsittelee laajemmin sitä, miten sanavektoreita käytetään luonnollisen kielen esittämiseen puheentunnistuksessa; luvussa 5 tuodaan esiin matemaattiset mallit ja niiden hyödyntäminen puheentunnistuksessa; luvussa 6 määritellään syväoppiminen ja kerrotaan kuinka se on kasvattanut suosiotaan puheentunnistuksen alalla; luvussa 7 pohditaan syitä tälle suosion kasvulle syväoppimisen hyvien puolten ja heikkouksien esittelyllä ja luvussa 8 tehdään johtopäätöksiä aikaisempien lukujen perusteella.

2 Taustaa puheen tuottamisesta ja havaitsemisesta

Ihmisillä puheen tuottaminen perustuu kolmeen mekanismiin: initiaatioon, fonaatioon ja artikulaatioon. Initiaatiolla tarkoitetaan puheäänten tuottamisen voimanlähdettä, joka useimmiten on keuhkoista lähtevä uloshengitysilmavirta, mutta suomen kielen puhujille on ominaista tiettyjen äänteiden tuottaminen myös sisäänhengitysilmavirralla. Fonaatiolla taas tarkoitetaan tämän ilman saattamista aaltoliikkeeseen kurkunpäässä sijaitsevan ääniraon avulla. Yleisemmät fonaatiotyypit ovat soinnillinen (fonaatio tapahtuu) ja soinniton (fonaatio ei tapahdu). Muita mahdollisia tyyppejä ovat narina, kuiskaussointi ja henkäyssointi. (Laver, 2012.) Fonaation tuloksena syntyvä aalto ei säteile suoraan ulos, vaan se etenee nielun läpi kurkunpään yläpuolella ja sitten kielen yläpuolella olevan suuontelon läpi ja mahdollisesti nenäontelon läpi. Näitä polkuja kutsutaan äänikanavaksi ja nenäkanavaksi. Näissä kanavissa kulkevan ilmavirran kulun manipulointia kutsutaan artikulaatioksi. (Pulkki, 2015, s. 79–83.)

Suosituimmaksi tavaksi edellä mainittujen puheäänten transkriptioon eli niiden muuttamiseen kirjalliseen muotoon ja luokitteluun on pitkän kehityksen tuloksena noussut niin kutsuttu kansainvälinen foneettinen aakkosto (engl. International Phonetic Alphabet), johon usein viitataan kirjallisuudessa sen englanninkielisen nimen lyhenteellä IPA. IPA on tarkekirjoitusjärjestelmä, jonka tarkoituksena on esittää puhe tekstimuodossa ääntämyksen mukaan tieteellisen tarkasti siten, että jokaisella erilliseksi tunnistetulla äänteellä on oma symbolinsa. Näitä symboleita voidaan myös muokata edelleen liittämällä niihin erilaisia diakriittisiä merkkejä. IPA:ssa konsonantit ja vokaalit luokitellaan kahteen eri taulukkoon. Konsonantit luokitellaan artikulaatiopaikan, artikulaatiotavan ja soinnillisuuden mukaan. Vokaalit taas luokitellaan väljyyden, etisyyden/takaisuuden ja pyöreyden mukaan, jossa väljyys kuvaa kielen ja kitalaen välin suuruutta, etisyys/takaisuus kyseisen välin sijaintia suussa ja pyöreys huulten asentoa ääntämishetkellä. (Jones & Knight, 2015.)

Ääniaalto tai värähtely etenee fyysisessä väliaineessa, se voi vahvistua resonanssin vaikutuksesta ja vaimentua sellaisten häviöiden vuoksi, jotka muuttavat sen muihin energiamuotoihin. Resonanssi on ilmiö, jota esiintyy usein fysikaalisissa järjestelmissä. Signaalinkäsittelyn näkökulmasta ääniväylä ja nenäontelo toimivat suodattimena, joilla on säädettävät resonanssit, jotka korostavat tiettyjä taajuuksia. Ihmisäänen tapauksessa alinta taajuutta kutsutaan perustaajuudeksi, jota usein merkitään kirjallisuudessa termillä f0. Muita resonanssitaajuuksia kutsutaan formanteiksi. (Pulkki, 2015, s. 15–23.)

Ensiaskeleet äänen tallentamiseen ja esittämiseen visuaalisessa muodossa tehtiin 1900-luvun alkupuolella kokeellisen fonetiikan tutkimuksessa hyödyntämällä ns. kymografia, joka tallentaa paineenvaihteluita välittämällä ne suukappaleelta letkua pitkin värähtelevälle kalvolle, johon kiinnitetty piirrin rekisteröi liikkeen noetulla paperilla päällystetyn pyörivän rummun pinnalle. Eräs tärkeimmistä keksinnöistä äänen analysoinnissa tehtiin 1940-luvulla kehitetyn äänispektrografin myötä. Se oli alkumuodossaan menetelmä, jossa korkeintaan 2,4 sekunnin mittainen magneettinauhalle tallennettu puhenäyte syötettiin useita kertoja peräkkäin suodattimeen, jonka taajuusvastetta muutettiin manuaalisesti syöttökertojen välillä. Tämän jälkeen suodattimesta saatu signaali vahvistettiin ja poltettiin paperille. Kun koko ennalta määritelty taajuusalue oli näillä toistoilla käsitelty, tuloksena saatiin kaaviokuva, jossa vaaka-akseli esittää aikaa, pystyakseli äänen taajuutta ja paperilla olevan värin tummuus äänen amplitudia. Spektrografi erillisenä koneena on nykyään jo jäänyt historiaan, mutta itse analyysiperiaate muodostaa toiminnallisen pohjan nykyisille tietokoneella käytettäville puheanalyysiohjelmistoille ja spektrografikuvien tulkinta on edelleen tärkeä osa puheanalyysiä. (Jones & Knight, 2015.)

Vielä 1990-luvun puolivälissä akustinen puheanalyysi oli käytännöllisessä mielessä mahdollista ainoastaan isolla laitteistobudjetilla toimivilla ja teknisellä tuella varustetuissa laboratorioissa, mutta nykyään samat analyysit voi tehdä henkilökohtaisella tietokoneella. Useimpia käyttötapauksia varten ei ole enää tarpeellista käyttää kalliita, yksinomaan tiettyihin tarkoituksiin tehtyjä laitteistoja, vaikkakin laadukkaita ulkoisia äänikortteja suositetaan edelleen sisäänrakennettujen äänikorttien sijaan. Monikäyttöisiä analyysiohjelmistopaketteja, kuten esimerkiksi Sensimetricsin Speechstation2:ta on edelleen myynnissä, mutta Praat, ohjelmisto, josta on tähän päivään mennessä muodostunut puheanalyysissä alan standardi, on internetistä ladattavissa ilmaiseksi, toimii monilla käyttöjärjestelmillä ja on joustavampi ja sisältää enemmän toimintoja kuin myynnissä olevat ohjelmat. Praatia voi myös laajentaa tarpeiden mukaan liitännäisohjelmilla, joista esimerkkinä mainittakoon Akustyk, jota käytetään vokaalien analysointiin ja kaaviomuodossa esittämiseen. (Jones & Knight, 2015.)

Myös muita ilmaiseksi ladattavia ohjelmia on saatavilla ja monia näistä luonnehditaan Praatia käyttäjäystävällisemmiksi ja käyttöliittymältään intuitiivisemmiksi, mutta ne eivät ole saavuttaneet samaa suosiota kuin Praat. Koska analyysiohjelmat käyttävät erilaisia algoritmeja esimerkiksi sävelkorkeuden talteenottoon ja formanttien kartoittamiseen, niiden antamat tulokset eivät useinkaan ole yhteneviä, vaikka analysoitava näyte olisi täysin sama.

Tämän vuoksi tutkijoiden täytyy tuloksia raportoidessaan ilmoittaa millä analyysiohjelmalla ja millä kyseisen ohjelman versiolla tulokset on saatu. (Jones & Knight, 2015.)

Puhetta analysoitaessa voidaan olla tilanteessa, jossa analysoitava nauhoitus on tehty valvotuissa olosuhteissa parhaalla mahdollisella laitteistolla, parhaassa tapauksessa jopa itse analyysin tekijän valvonnassa, mutta useimmissa tapauksissa kyseistä materiaalia ei ole nauhoitettu parhaissa mahdollisissa olosuhteissa. Tapauksissa, jossa ollaan kentällä tai muuten laboratorion ulkopuolella nauhoituksia ei voida tehdä hiljaisessa ja kontrolloidussa ympäristössä. Puheen tallentamiseen käytettyjen mikrofonien ja tallennuslaitteiden laatu vaihtelee suuresti ja tuloksena saadun nauhoituksen tekniset ominaisuudet vaihtelevat sellaisten tekijöiden kuin mikrofonin tyypin (esimerkkinä tallentimen sisäänrakennettu mikrofoni vs. esivahvistimeen kytketty ulkoinen monisuuntainen rajamikrofoni) ja sen taajuusvasteominaisuuksien mukaan. Mitä matalampi taajuusvasteen spektri on, sitä enemmän nauhoitus vastaa alkuperäistä ääntä. Tallentimen näytteenottotaajuus ja bitin syvyysasetus, se nauhoitetaanko mono- vai stereoääntä ja valittu tiedostomuoto (häviötön pulssikoodimodulaatio (PCM) vs. häviöllinen, pakattu muoto, kuten .mp3) määrittävät myös nauhoitetun signaalin laadun. (Jones & Knight, 2015.)

3 Puheen muuttaminen ja tallentaminen digitaaliseen muotoon

Analogisen ääniaallon muuttamisessa digitaaliseksi binäärikoodiksi kohdataan väistämättä se ongelma, että kyseessä on kaksi hyvin erityyppistä informaation muotoa. Analogisessa muodossa ääniaalto on luonteeltaan jatkuvaa, jolloin epäjatkuvuudet useimmiten mielletään vakaviksi virhetilanteiksi. Digitaalinen informaatio on jakautuessaan bitteihin ja tavuihin perustavanlaatuisesti epäjatkuvaa. Tällä on sekä hyviä että huonoja puolia. Kyky pilkkoa analoginen signaali pienemmiksi paloiksi on erittäin hyödyllinen äänisignaalin siirtämisessä ja tallettamisessa. Mutta ellei datan kokoamista tehdä kunnolla, analogisessa äänisignaalissa olevat vääristymät, jotka eivät siinä muodossa haittaa viestin ymmärtämistä, voivat digitaalisessa muodossa tehdä viestistä täysin tunnistamattoman. (Davis ym., 2013.)

Äänisignaalien muuntamisessa tärkeä käsite on ns. LTI-järjestelmä. Järjestelmä on lineaarinen ja aikainvariantti (engl. Linear and Time-Invariant, LTI), jos järjestelmän vasteen kahden tulosignaalin summa on yhtä suuri kuin yksittäisten tulojen vasteiden summa erikseen. LTI-järjestelmien analysointi ja toteutus on tyypillisesti helpompaa ja tehokkaampaa kuin sellaisten järjestelmien, joilla ei ole tätä ominaisuutta. Valitettavasti ihmiskuulo on hyvä esimerkki järjestelmästä, joka ei ole pohjimmiltaan LTI-järjestelmä. LTI-järjestelmän ominaisuutena on myös se, että se ei luo uusia taajuuskomponentteja, joita ei ole tulosignaalissa, jolloin alkuperäisen ja muunnetun äänen vastaavuudesta voidaan olla varmempia. (Pulkki, 2015, s. 46–47.) LTI-järjestelmien merkitys sujuvalle puheentunnistukselle on siinä, että erityisesti reaaliajassa suoritettava puheentunnistus edellyttää sitä, että äänimateriaalille suoritettavat laskutoimitukset eivät ole liian monimutkaisia, jolloin säästetään laskentatehoa. Tämän ja ihmiskuulon luonteen vuoksi puheen muuntamisessa joudutaan tyytymään kompromisseihin.

Signaalinkäsittelyssä on usein hyödyllistä signaalien muuntaminen sellaiseen muotoon, joka helpottaa jollain tavalla niiden käsittelyä tai tulkintaa. Esimerkki tällaisesta hyödyllisestä muunnoksesta on saman signaalin muuntaminen aika-alueen esityksestä taajuusalueen esitykseen, joka LTI-järjestelmällä tekee analyysistä matemaattisesti yksinkertaisemman. Tämä voidaan tehdä käyttämällä Fourier-muunnosta, mutta käytännön sovelluksissa Fourier-muunnos tehdään käyttämällä ns. nopeaa Fourier-muunnosta (engl. Fast Fourier Transform, FFT). Syy tähän on se, että määritelmän mukaisen Fourier-muunnoksen aikavaativuus on neliöllinen, kun yleisimmin FFT:een käytetyllä Cooleyn-Tukeyn algoritmilla aikavaativuusluokka on O(N log N). (Pulkki, 2015, s. 49.)

Äänisignaalin muuntamisessa käytetään spektrianalyysiä tekemällä ikkunointi, jossa signaali kerrotaan ikkunafunktiolla ja tästä tuloksesta tehdään Fourier-analyysi. Ikkunoinnissa käytetyt muuttujat ovat ikkunafunktio ja näytteenotossa käytettävän aikaikkunan kesto. Esimerkkejä ikkunafunktioista ovat Hammingin, Hannin, Blackmanin ja Kaiserin ikkunafunktiot. Useimmissa sovelluksissa aikaikkunan kesto on 10–30 millisekuntia, puheen analyysissä 10 millisekuntia on tavallisin. Edellisessä luvussa mainittu 1940-luvulla kehitetty äänispektrografia on varhainen käytäntöön sovellettu versio tästä prosessista, jonka graafisena esityksenä saadaan spektrogrammikuvio. (Pulkki, 2015, s. 50–53.)

Analogiset aaltomuodot muunnetaan digitaaliseksi analogi-digitaalimuuntimella ottamalla jännitenäytteitä tietyllä aikavälillä. Käytettävissä olevien jännitearvojen lukumäärä, joka voidaan osoittaa kullekin näytteelle, on noin 2N, missä N on kutakin näytettä edustavien bittien lukumäärä. Mitä enemmän bittejä, sitä enemmän on saatavilla olevia arvoja ja sitä suurempi on lopputuloksen dynaaminen alue. Nyquistin periaatteen mukaan näytteenottotaajuuden on oltava hieman yli kaksi kertaa korkeampi kuin aaltomuodossa oleva korkein taajuus, jotta kaikki sen sisältämä ääni-informaatio voidaan saada talteen. Antialiasointisuodatinta käyttämällä pystytään automaattisesti hylkäämään taajuudet, joille valittu näytteenottotaajuus ei ole riittävä. (Davis ym., 2013.)

Yksi yleisimmistä tavoista tallentaa puhetta tietokoneen muistiin on käyttää jo edellisen luvun lopussa mainittua pulssikoodimodulaatiota (PCM). PCM-järjestelmä, joka tunnetaan myös edellä mainittuna analogi-digitaalimuuntimena, koostuu kolmesta komponentista: näytteenottimesta, kvantisaattorista ja enkooderista. Näytteenotin pilkkoo äänisignaalin aiemmin tässä luvussa kuvatulla tavalla, jolloin tuloksena on diskreetin ajan ja jatkuvan amplitudin signaali. Kvantisaattorilla jokaisen näytteen amplitudi pyöristetään sallitulle tasolle, jolloin tuloksena on signaali, jolla on diskreetti aika ja diskreetti amplitudi. Enkooderilla jokainen kvantisaattorilta tuleva näyte kuvataan sille ominaisen tason mukaan tietyllä binäärimuotoisella luvulla. (Bhagyaveni ym., 2016, s. 98–104.) PCM-koodattua ääntä käytetään esimerkiksi WAVE:ssa (Waveform Audio File Format), joka on eräs tärkeimmistä häviöttömistä tiedostomuodoista äänen tallentamiseen tietokoneen muistiin. WAVE-muotoinen äänitiedosto sisältää PCM-äänen lisäksi myös infodataa, joka kertoo mm. näytteenottotaajuuden lohkoina sekunnissa, yksittäisen lohkon koon tavuina ja bittien määrän näytettä kohden. (IBM & Microsoft, 1991.)

4 Sanavektorien käyttö luonnollisen kielen esittämiseen

Puheentunnistuksessa, kuten muissakin luonnollisten kielten käsittelyjärjestelmissä, törmätään ongelmaan, jossa yksittäisen sanan merkityksellä ja sen kirjoitusmerkkimuotoisella esityksellä ei ole suoraa yhteyttä toistensa kanssa, mutta toisaalta kahden kirjoitusmerkein esitettynä identtisen sanan merkitys voi olla täysin erilainen riippuen siitä minkä muiden sanojen yhteydessä kyseinen sana esiintyy (Wang ym., 2020). Ihmiset kykenevät kuitenkin luontaisesti mieltämään tietyt sanat samankaltaisemmiksi kuin toiset sanat, miten haluaisimme myös käsittelyjärjestelmän toimivan. Sanojen ja niiden merkitysten yhdistämistä tietokoneen ymmärtämään muotoon on jo kyseisten järjestelmien kehityksen alkuajoista asti pyritty tekemään vektorimuotoisina sananupotuksina (engl. word embeddings), joihin yleisesti viitataan sanavektoreina (Mitkov, 2014, s. 334–358). Ensimmäiset sanavektorit olivat suurikokoisia one-hot-koodattuja vektoreita, mikä tarkoittaa sitä, että vektorin alkioiden määrä on yhtä suuri järjestelmän käyttämän sanavaraston kanssa ja oikeaa merkitystä ilmaiseva alkio on koodattu ykköseksi muiden alkioiden ollessa nollia. Tästä seuraa se, että kaikkien sanavektorien yhdessä muodostama matriisi on kooltaan sanavaraston suhteen neliöllinen, mikä on matriisin sisältämään informaatioon verrattuna paljon tilaa vievä tapa merkitysten tallettamiseen. Kaikki sanavektorit ovat myös suorassa kulmassa toisiinsa nähden, jolloin kaikki sanat ovat myös yhtä samankaltaisia toistensa suhteen. (Wang ym., 2020.)

Sanojen samankaltaisuuden mallintamiseen tarvitaan enemmän ominaisuuksia, joiden kehyksenä käytetään Harrisin (1954) ja Firthin jakautumishypoteesia (engl. Distributional Hypothesis). Jakautumishypoteesin mukaan kahden sanan merkitysero korreloi niiden kontekstien jakautumisen eron kanssa. Käytännössä tämä voidaan toteuttaa niin, että sanoista ja niihin liittyvistä konteksteista muodostetaan erillinen matriisi, jota käytetään yhdessä edellisessä kappaleessa kuvatus sanavarastomatriisin kanssa. (Wang ym., 2020.) Sanavarastomatriisin ja kontekstimatriisin välisillä laskutoimituksilla voidaan mitata sanojen ja niiden kontekstien välistä korrelaatiota, josta jakautumishypoteesin mukaisesti hyödyntäen erilaisia painotuksia ja matriisien manipulointia sanojen semanttinen samankaltaisuus voidaan määritellä etäisyytenä vektoriavaruudessa. Tärkeä välivaihe näissä laskutoimituksissa on ns. samanaikaisen esiintymisen matriisi (engl. co-occurence matrix), jossa matriisin rivillä olevan sanan ja sarkkeessa olevan kontekstin leikkauskohdassa on kyseisen sana-konteksti-parin esiintymien määrä tutkittavassa aineistossa. (Mitkov, 2014, s. 334–358.)

On olemassa useita eri kontekstityyppejä, joita voidaan käyttää matriisien muodostamiseen. Eräs vanhimmista konteksteista oli selvittää, esiintyykö sana tietyssä tekstidokumentissa ja vaihtoehtoisesti myös, kuinka monta kertaa kyseinen sana esiintyy. Vielä tälläkin hetkellä yleisin käytössä oleva konteksti on selvittää, mitä sanoja tarkasteltavan sanan naapureina tekstissä on tietyn välimatkan päässä sekä sanan edellä että perässä olevan ikkunan sisällä. Muita konteksteja ovat syntaktisten riippuvuuksien tai symmetristen kuvioiden muodostamat yhteydet tai sanayhdistelmät. Kontekstien ei tarvitse muodostua pelkästään sanoista, vaan myös kuvia on mahdollista hyödyntää niiden tekemiseen. (Mitkov, 2014, s. 334–358.)

Sanavektoreita voidaan käyttää syötteenä syville hermoverkolle. Riippuen siitä, mitä mallia käytetään, ne voidaan tulkita joko vakioiksi tai parametreiksi. On myös yleistä, että syväoppimismalleilla koulutettuja sanavektoreita käytetään syötteenä muille malleille. (Mitkov, 2014, s. 334–358.)

5 Koneoppiminen ja sen soveltaminen puheentunnistukseen

Koneoppiminen määritellään tekoälyn osa-alueeksi, jolla kuvataan niitä tietokoneille kehitettyjä työkaluja, jotka mahdollistavat syöttödatasta oppimisen ilman, että sitä on erikseen ohjelmoitu (Gerard, 2021). Nassifin ja muiden (2019) artikkelin mukaan ”oppimisprosessi tapahtuu iteratiivisesti analysoidusta datasta ja uudesta syöttödatasta”. Tämän jälkeen he mainitsevat, että ”tämä antaa tietokoneille mahdollisuuden tunnistaa piileviä oivalluksia ja toistuvia kuvioita ja käyttää näitä havaintoja sopeutuakseen, kun ne altistuvat uudelle datalle”. Oppimisprosessin lopputuloksena saadaan malli. Koneoppimisessa mallit ovat matemaattisia funktioita, jossa uudet syötteet esitetään parametreinä ja jonka tuloksena saadaan ennuste (Gerard, 2021). Viisi koneoppimisen päätekniikkaa ovat: ohjattu oppiminen, ohjaamaton oppiminen, puoliohjattu oppiminen, vahvistusoppiminen ja syväoppiminen (Nassif ym., 2019).

Ohjatussa oppimisessa käsiteltävästä datasta muodostetaan harjoitusjoukko, joissa havainnot esitetään pareittain syötteenä ja sitä vastaavana tulosteena. Oppimista kutsutaan ohjatuksi, koska oikea tulos tiedetään ja mallin muodostavan algoritmin ennustamien tulosten ja oikeiden tulosten välistä eroa pyritään pienentämään ohjaamalla oppimisprosessia sen etenemisen aikana. Ohjaamaton oppiminen puolestaan yrittää löytää yhteisiä piirteitä havaintojen ominaisuuksista. Mikäli jokin ominaisuus korreloi huomattavan havaintomäärän kanssa tilastollisten ominaisuuksiensa perusteella, kyseinen ominaisuus näkyy kuvaajassa lähekkäin olevien havaintojen joukkona. Puolivalvottu oppiminen on kahden aiemmin kuvatun tyypin yhdistelmä, jossa algoritmia koulutetaan käyttämällä tietojoukkoa, joka sisältää sekä luokiteltuja että luokittelemattomia syötteitä. Vahvistusoppiminen perustuu agentteihin, jotka suorittavat toimintoja valitsemalla niitä niille annetuista vaihtoehdoista. Jokaisella vaihtoehdolla on numeerinen painoarvo, jonka summan agentti pyrkii maksimoimaan. Näin pyritään siihen, että algoritmi ”ymmärtäisi”, mikä on tavoitteen saavuttamiseksi paras toimintojen sarja. (Nassif ym., 2019.)

HMM-malli on määritelmän mukaan kahdesti stokastinen prosessi, jossa on taustalla erillinen stokastinen prosessi, joka ei ole suoraan tarkasteltavissa. Tästä prosessista voidaan tehdä kuitenkin päätelmiä epäsuorasti toisen stokastisen prosessin, joka tuottaa sarjan tarkasteltavia symboleja, kautta. (Rabiner & Juang, 1986.) Stokastinen prosessi on tässä tapauksessa yksinkertaisesti sarja satunnaisia muuttujia (Mitkov, 2014). Jokaisella sarjassa olevalla muuttujalla on rajallinen määrä tiloja ja sarjan ensimmäistä muuttujaa lukuun ottamatta

jokaisen muuttujan tila voidaan kuvata siirtymänä edellisen muuttujan tilasta, mikä toteutuu tietyllä todennäköisyydellä kaikkien mahdollisten tilojen muodostaman tila-avaruuden sisällä. Tätä mekanismia kutsutaan Markovin prosessiksi. Kieliteknologisessa mielessä kielen sanavarasto muodostaa tällaisen tila-avaruuden ja lauseet voidaan tulkita sarjaksi muuttujia, joita yksittäiset sanat ovat. Malli pitää tulkita piilotetuksi, koska voimme tehdä päätelmiä sanojen yhteyksistä toisiinsa ainoastaan tuotetun kielen pohjalta. (Rabiner & Juang, 1986.)

Song (2019) kuvaa artikkelissaan HMM-mallin matemaattisen periaatteen, kun sitä hyödynnetään puheentunnistuksessa. Annetulle puhesignaalin akustiselle piirrevektorisarjalle $O^1T = \{o_1, o_2, \ldots, o\}$ tehdään dekoodaushaku yhdistämällä akustinen malli ja kielimalli. Todennäköisintä sanasarjaa kuvataan merkinnällä $W^* = \{w_1, w_2, \ldots, w_n\}$. Tällöin puheentunnistuksen prosessi voidaan kuvata maksimointiongelmaksi, jossa yritetään maksimoida posteriorinen todennäköisyys $P(W \mid O^1T)$, joka löydetään maksimoidun posteriorisen todennäköisyyden kriteereillä kaavalla:

$$W^* = argmax\{\frac{P(O_T^1|W)P(W)}{P(O_T^1)}\}$$

Yläpuolella olevassa yhtälössä P(W) kuvaa kielimallin todennäköisyyttä, jolla viitataan todennäköisyyteen, jolla sanasarjan W yksittäinen esiintymä voidaan tunnistaa. Tämä todennäköisyys on riippumaton sarjasta O^1T. Koska kyseisen sarjan koko ei muutu yksittäisen havainnon tapauksessa, kaava voidaan sieventää muotoon:

$$W^* = argmax\{P(O_T^1|W)P(W)\}$$

Tämä tunnetaan puheentunnistuksen yleiskaavana. Sillä pyritään löytämään optimaalinen sanasarja W*, joka maksimoi laskutoimituksen tuloksen.

Poiketen luvussa 1 mainitusta HMM- ja GMM-mallien käytön eroista puheen eri ominaisuuksien käsittelyssä, Nassif ja muut (2019) kuvaavat näiden mallien yhteyttä artikkelissaan myös siten, että puhesignaalia voidaan pitää Markovin prosessin mukaisena lyhytaikaisena stationaarisena signaalina, mutta samalla HMM-malli pitää sisällään GMM-mallilla toteutetun ääniaallon spektrimuotoisen esityksen.

6 Syväoppiminen

Chen ja Lin (2014) määrittelevät artikkelissaan syväoppimisen ”koneoppimistekniikoiksi, jotka käyttävät ohjatun ja/tai ohjaamattoman oppimisen menetelmiä hierarkkisten esitysten automaattiseen oppimiseen syvissä verkkoarkkitehtuureissa”. He jatkavat mainitsemalla, että syväoppimisen toteutus on ”saanut inspiraationsa biologisista havainnoista, kuinka ihmisaivojen mekanismit käsittelevät luonnollisia signaaleja”. Kiinnostuksesta syväoppimista kohtaan he lisäävät, että ”yritykset, kuten Google, Apple ja Facebook, jotka keräävät ja analysoivat valtavia määriä dataa päivittäin, ovat aggressiivisesti ajaneet eteenpäin syväoppimiseen liittyviä projekteja”.

Kaksi erityisesti puheentunnistuksessa käytettävää syväoppimismallia ovat toistuvat hermoverkot (engl. Recurring Neural Network, RNN) ja RNN-malliin pohjautuva LSTM-malli (Long Short Term Memory). RNN-mallin käyttö puheentunnistuksessa perustuu syntaktisen systemaattisuuden ja rekursion periaatteisiin. Syntaktisessa systemaattisuudessa sanan korvaaminen toisella samaan leksikaaliseen luokkaan kuuluvalla sanalla (esim. substantiivin korvaaminen toisella substantiivilla) ei muuta lausetta kieliopin vastaiseksi. Rekursio tarkoittaa tässä tilanteessa sitä, että lauseen syntaksi mallinnetaan säännöillä, jotka määrittävät osittain itse itsensä. Puheentunnistuksessa RNN-malli voi yksinkertaisimmillaan koostua kahdesta syötekerroksesta, kontekstikerroksesta, piilokerroksesta ja tulostekerroksesta, jossa toistuva yhteys on piilokerroksen ja kontekstikerroksen välillä. (Sakurai & Shinozawa, 2008.) LSTM-malli on toistuva hermoverkko, jonka erityispiirre on se, että se koostuu useista pienemmistä muistisoluiksi kutsutuista yksiköistä, jotka muistuttavat rakenteeltaan hyvin yksinkertaisia hermoverkkoja. Muistisolut pystyvät pitämään sisällään niille syötettyä dataa, päästämään sen ulos tai tyhjentämään oman sisältönsä tiettyjen ehtojen toteutuessa. LSTM-malli on todettu käyttökelpoiseksi myös muun aikasarjamuotoisen datan mallinnuksessa ja sitä on hyödynnetty myös automaattisessa tekstitysten luonnissa. (Van Houdt ym., 2020.)

Tärkeä edistysaskel syväoppimisen käytön houkuttelevuudelle on ollut erilaisista digitaalisista lähteistä tulevan datan määrän ja tuottonopeuden valtava kasvu. Perinteisillä analyysimenetelmillä, jotka vaativat ihmistyönä tehtävää datan valmistelua, tästä datasta pystyttäisiin käsittelemään tämän valmistelun aikaa vievän luonteen vuoksi vain pieni osa, joka luultavasti tulisi datan määrän ja nopeuden, jolla uutta dataa syntyy, edelleen kasvaessa pienenemään entisestään. Syväoppimisessa on tämän vuoksi hyödynnetty enemmän

ohjaamattoman oppimisen menetelmiä, joiden on todettu hyötyvän mallien koon ja datamäärän kasvattamisesta. Mallien koon kasvattamisesta syntyviä ongelmia on pyritty ratkaisemaan kehittämällä rinnakkaislaskentaan perustuvia järjestelmiä, jotka toimisivat syväoppimismallien kanssa. Rinnakkaislaskennalla pystytään myös hyödyntämään moniytimisiä suorittimia. (Chen & Lin, 2014.)

7 Syväoppimisen edut tilastollisiin malleihin verrattuna

Syvien hermoverkkojen parempi suorituskyky perinteisiin matemaattisiin malleihin verrattuna perustuu Dengin ym. (2013) mukaan pääasiallisesti kolmeen tekijään: verkon kerrosten määrän kasvattamiseen, malleissa käytettävien parametrien painotusten tarkoituksenmukaisempaan käyttöön ja tulostusten määrän kasvattamiseen. Myöhemmin tehdyt parannukset ovat olleet seurausta painotusten suuruusluokan optimoinnista, puhujasta riippumattomien menetelmien yleistymisestä, konvoluutiokerrosten hyödyntämisestä ja moniajosta.

Pouyanfar ym. (2019) esittävät artikkelissaan, että syväoppimismallit ovat suorituskykyisempiä kuin perinteiset matemaattiset mallit. He perustelevat tätä sillä, että perinteisten koneoppimisalgoritmien tehokkuus on riippuvainen, kuinka hyvin niille syötetty data on esitetty. Tämän vuoksi ominaisuuksien rakentamiseen raakadatasta on keskeistä siihen, että malleista saadaan käyttökelpoisia tuloksia, mutta tämä vaatii usein paljon ihmistyötä. Syväoppimisalgoritmit pystyvät poimimaan ominaisuuksia automaattisesti, minkä vuoksi mallinnuksen voi tehdä ilman erityisosaamista kyseisestä tutkimusalueesta ja pienemmällä vaivannäöllä. Näillä algoritmeilla korkean tason ominaisuudet voidaan poimia verkon viimeisistä kerroksista, kun taas matalan tason ominaisuudet voidaan erottaa jo aikaisemmista kerroksista. Tässä artikkelissa luetellaan seitsemän eri syväoppimisverkkoa: rekursiiviset hermoverkot, toistuvat hermoverkot, konvoluutiohermoverkot sekä syvistä generatiivisista hermoverkoista DBN-verkko, Deep Boltzmann Machine (DBM), generatiivinen vastavuoroinen verkosto (engl. Generative Adverserial Network, GAN) ja Variational Autoencoder (VAE). He kuitenkin toteavat myös sen, että syväoppiminen on tutkimusalana kasvanut erittäin nopeasti ja monia uusia verkkoja ja uusia arkkitehtuureja ilmestyy muutaman kuukauden välein ja siksi ovat joutuneet jättämään tämän uusimman kehityksen tutkimuksen ulkopuolelle.

Nassif ym. (2019) tekivät artikkelissaan systemaattisen kirjallisuuskatsauksen tutkimusartikkeleista, joissa käsiteltiin syväoppimisen hyödyntämistä puheentunnistuksessa. Tähän katsaukseen valikoitiin 174 artikkelia, jotka oli julkaistu vuosina 2006–2018. Nämä artikkelit arvioitiin kahdeksan eri tutkimuskysymyksen avulla, jotka olivat:

- Minkä tyyppisistä artikkeleista oli kyse?
- Minkä tyyppistä puhetta artikkeleissa käsiteltiin?

- Minkä tyyppisiä tietokantoja käytettiin algoritmien testaukseen ja opetukseen?
- Mitkä olivat artikkeleissa käytetyt tietokantojen kielet?
- Millaisessa ympäristössä tutkimus toteutettiin?
- Kuinka ominaisuudet otettiin talteen puheesta?
- Mitä arviointitekniikoita artikkeleissa käytettiin?
- Minkä tyyppisiä syväoppimismalleja käytettiin?

Näiden kysymysten pohjalta katsauksessa päädyttiin lukuisiin johtopäätöksiin. Suurin osa (40 %) artikkelista oli konfrenssijulkaisuja ja yli 50 % niistä julkaistiin ICASSP:n (International Conference on Acoustics, Speech, and Signal Processing) yhteydessä. Suurin osa artikkeleista perustui julkisiin, englanninkielisiin tietokantoihin ja suurimassa osassa tutkimusympäristö oli luonnollinen ja taustameluton. Suurin osa artikkeleista käytti sanojen virheprosenttia mallin suorituskyvyn määrittelyyn. Yllättävänä pidettiin sitä, että suurimmassa osassa artikkeleita käytettiin edelleen MFCC:tä (Mel-frequency cepstral coefficients, suom. Mel-taajuuksien kepstrikertoimet) puheen ominaisuuksien talteenottoon, vaikka sitä on hyödynnetty jo pitkään HMM:en ja GMM:en yhteydessä. Syväoppimismalleissa olisi kannattavaa hyödyntää muita tapoja ottaa ominaisuuksia talteen, kuten esimerkiksi LPC:tä (Linear Predictive Coding). Noin 75 % artikkeleissa käsitellyistä malleista oli yksittäisiä syviä hermoverkkomalleja ja loput hybridimalleja. Kirjoittajat viittasivat tässä vaiheessa vielä tällöin julkaisemattomaan artikkeliin, jossa oli osittain samoja tekijöitä kuin tässä kirjallisuuskatsauksessa, kun he suosittelivat yleisemmin hybridimallien käyttämistä pelkkien syvien neuroverkkojen tai Gaussin sekoitusmallien sijasta. Tämä suositus perustui kuitenkin vain tähän yhteen Shahinin ym. (2020) artikkelissa suoritettuun tutkimukseen, jonka tutkimusasetelma perustui siihen, miten puhujien eri tunnetilat vaikuttavat puheentunnistuksen suorituskykyyn. Tämän tutkimuksen tuloksiksi saatiin jokaisessa tapauksessa se, että GMM-DNN-tyypin hybridimallin suorituskyky oli parempi kuin GMM- tai DNN-mallin suorituskyky. DNN-mallin suorituskyky oli kuitenkin jokaisessa tapauksessa parempi kuin GMM-mallin suorituskyky. Johtopäätöstensä lopuksi kirjoittajat huomioivat sen, että toistuvia hermoverkkoja oli käytetty suhteellisen vähän tutkimusmenetelmänä ja suosittelivat sen lisäämistä, koska heidän mielestään RNN:in kuuluvista malleista erityisesti LSTM-malli on puheentunnistuksessa hyvin tehokas.

Song (2020) vertasi omassa tutkimuksessaan kolmen eri mallin suorituskykyä. Kaksi niistä olivat jo aiemmin mainitut HMM-malli ja GMM-HMM -hybridimalli. Kolmantena mallina käytettiin CNN-RBM-ASAT -hybridimallia, joka koostuu lueteltujen lyhenteiden mukaisesti konvoluutiohermoverkosta ja rajoitetusta Boltzmann-koneesta (engl. restricted Boltzmann machine). Kolmantena osapuolena tässä mallissa on ASAT (Automatic Speech Attribute Transcription), joka toimii mallin muita osia seuraavana käsittelyjärjestelmänä, joka käyttää edellisestä mallista tulosteena saatuja parametreja syötteenä ennalta määrättyihin ominaisuusluokittelijoihin (Hou ym., 2006). Tutkimuksessa käytetyn CNN-RBM-ASAT -mallin rakennetta kuvattiin siten, että se sisältää yhden syöttökerroksen, viisi piilokerrosta ja yhden tuloskerroksen. Tämä tulos syötettiin sitten 21 ominaisuuskategoriaa sisältävälle ASAT-luokittelijalle. Suorituskyky mitattiin puheen attribuuttien oikein tehtyjen tunnistusten määränä. Aikaisemmin mainitun kolmen mallin puheen ominaisuuksien tunnistusvertailussa CNN-RBM-ASAT -malli suoriutui useimpien ominaisuuksien tapauksessa paremmin kuin muut mallit. Samassa tutkimuksessa vertailtiin myös CNN-RBM-ASAT -mallin sana- ja lausevirheiden määrää verrattuna CNN-HMM-, DNN-HMM-, ja CNN-BRM-mallien virhemääriin. Tulokseksi saatiin se, että CNN-RBM-ASAT -malli teki akustisessa mallinnuksessa vähemmän virheitä kuin muut vertailtavat mallit. Tästä voidaan edelleen tehdä se johtopäätös, että CNN-mallilla on vahvempi mallinnuskyky kuin DNN-mallilla monimutkaisella datalla ja sen mallinnuskyky on myös edullisempi. Analyysissä saatiin näiden tietojen lisäksi selville se, että mallin suorituskyky harjoitussarjassa on huomattavasti parempi kuin testisarjassa, koska malli on taipuvainen ylisovitukseen harjoitussarjassa ja testisarja voi estää tämän ja parantaa järjestelmää.

8 Johtopäätökset

Puheentunnistus on matemaattisten mallien ja syväoppimisen käytön kannalta monia tieteenlajeja yhdistelevä sovelluskohde. Fonetiikan avulla pystytään kartoittamaan se, kuinka yksittäisen sanan muodostavien äänteet voi tunnistaa niiden taajuusominaisuuksien perusteella ja millaiset taajuusmuutokset merkitsevät tiettyjä siirtymiä eri äänteestä toiseen. Näin saadaan se pohjatieto, mihin matemaattisiin malleihin syötettävää dataa verrataan. Signaalinkäsittelyä tarvitaan puhedatan muuttamiseen analogisesta muodosta digitaaliseen muotoon. Tämän muunnoksen laatu on keskeinen vaatimus onnistuneessa puheentunnistuksessa sen vuoksi, että toimiakseen mahdollisimman virheettömästi malleille syötettävän datan on oltava mahdollisimman hyvälaatuista.

Koneoppiminen ja syväoppiminen ovat molemmat yhdessä hermoverkkojen kanssa tekoälyn osa-alueita. Puheentunnistus on vain yksi monista tekoälyn sovelluskohteista esimerkiksi käsin kirjoitetun tekstin tunnistuksen, hakukoneiden ja konenäön ohella. Toisaalta syväoppimisen merkittävin ero koneoppimiseen verrattuna on käytetyn verkon kerrosten määrässä, syväoppimisverkoissa kerrosten määrä on huomattavasti suurempi kuin perinteisissä koneoppimisverkoissa. Tässä tutkielmassa käyttämissäni lähteissä viitataan kone- ja syväoppimisen eroihin niitä puheentunnistukseen sovellettaessa siten, että kummallakin tekniikalla on edelleenkin omat hyvät puolensa riippuen siitä, kuinka monimutkainen käyttökohde on. Esimerkiksi Chenin & Linin (2014) mukaan syväoppimisen edut pääsevät paremmin esille silloin, kun käsiteltävän datan määrä on hyvin suuri ja vähän kerroksia sisältävässä verkossa myös parametrien määrän pitää olla pieni ylisovitusongelmien välttämiseksi.

Syväoppimismalleille ominainen verkkokerrosten huomattavasti suurempi määrä on johtanut siihen, että niiden toiminta on vähemmän riippuvaista ominaisuuksien luokittelusta. Ominaisuusluokittelua ei käytännössä voi suorittaa tilanteissa, joissa dataa syntyy jatkuvana virtana ja sen käsittely on tehtävä hyvin pienellä viiveellä. Perinteisten koneoppimismallien suorituskyky taas perustuu siihen, että ominaisuuksien luokittelu on tehty huolellisesti. Syväoppimisessa voidaan käsin tapahtuvan luokittelun sijaan käyttää verkko-oppimista, jossa opitaan yksi ilmentymä kerrallaan ja malli tarkentuu iteratiivisesti datavirrasta poimittujen esiintymien mukaan. Tätä kutsutaan yleisesti ohjaamattomaksi oppimiseksi ja edellisen perusteella voidaan päätellä, että syväoppimismallit ovat rakenteensa vuoksi parempia hyödyntämään tätä oppimistapaa. Chen ja Lin (2014) kuitenkin mainitsevat, että

syväoppimismalleja voi käyttää sekä koulutusdatan avulla tapahtuvaan ohjattuun oppimiseen että ohjaamattomaan oppimiseen.

Syväoppimismallien eduksi perinteisiin koneoppimismalleihin verrattuna voidaan laskea se, että tekoälyn viimeaikainen tutkimustyö ja kehitys on tapahtunut sellaisilla alueilla, jotka ovat läheisesti yhteydessä syväoppimiseen. Esimerkkinä tästä voidaan mainita ominaisuussuunnittelu, jossa piirteet rakennetaan syötedatasta automaattisesti klusterointia hyödyntäen. Uusia syväoppimisverkkoja ja verkkoarkkitehtuureita on kehitetty myös hyvin säännöllisesti. Vaikka syväoppimismallit ovat yleisesti ottaen perinteisiä malleja tehokkaampia, joissain tutkimuksissa on todettu, että perinteisten mallien ja syväoppimismallien rakenteita yhdistelevät ns. hybridimallit ovat suorituskyvyltään tietyissä puheentunnistuksen sovelluksissa parempia kuin kumpikaan edellisistä malleista yksin. Hybridimallien vahvuuksiksi mainittiin pienemmän virhemäärän lisäksi myös parempi suorituskyky ja laskennallisella kuormittavuudella mitattuna kevyempi mallinnuskyky.

9 Yhteenveto

Tavallinen ihminen on yleensä puheentunnistuksen kanssa tekemisissä nykyään erilaisten puheenkäsittelysovellusten kanssa, joista huomattavimmiksi ovat aivan viime aikoina nousseet erilaiset virtuaaliavustajat. Näiden sovellusten taustalla oleva teoria periytyy jo tekoälytutkimuksen alkuajoilta 1950-luvulta, mutta vasta tietojenkäsittelyn suorituskyvyn kasvu on tehnyt mahdolliseksi teorian soveltamisen nykyisessä laajuudessaan.

Puheentunnistuksen prosessissa puhe ensin tallennetaan ja muutetaan digitaaliseen muotoon, jonka jälkeen digitoitu puhe pilkotaan pieniin paloihin, joista jokaisesta muodostetaan sen aaltomuodon sisältämän taajuusinformaation muodostama piirrevektori. Näitä piirrevektoreita verrataan tunnettujen äänteiden piirrevektoreihin, joista muodostetaan kyseisen puhenäytteen äännemalli. Äännemallin virheitä voidaan tunnistaa vertaamalla sitä kyseisen puhekielen kielimalliin, joka arvioi kuinka todennäköisiä tarkasteltavan äännemallin äännerakenteet ovat kohdekielessä. Fonetiikan avulla pystytään kartoittamaan se, kuinka yksittäisen sanan muodostavien äänteet voi tunnistaa niiden taajuusominaisuuksien perusteella ja millaiset taajuusmuutokset merkitsevät tiettyjä siirtymiä eri äänteestä toiseen. Signaalinkäsittelyä tarvitaan puhedatan muuttamiseen analogisesta muodosta digitaaliseen muotoon käyttäen hyväksi Fourier-muunnosta.

Koneoppiminen ja syväoppiminen ovat molemmat yhdessä hermoverkkojen kanssa tekoälyn osa-alueita. Koneoppiminen tarjoaa tietokoneille mahdollisuuden oppia syöttötiedoista ilman, että niitä on erikseen ohjelmoitu tekemään niin. Oppimisprosessi tapahtuu iteratiivisesti perustuen analysoituun dataan ja uusiin syöttötietoihin. Koneoppimistekniikat voidaan luokitella viiteen ryhmään, jotka ovat: ohjattu oppiminen, ohjaamaton oppiminen, puoliohjattu oppiminen, vahvistusoppiminen ja syväoppiminen. Esimerkkinä koneoppimisessa käytettävästä ns. perinteisestä matemaattisesta mallista voidaan käyttää piilotettua Markovin mallia (HMM). Kun HMM-mallia sovelletaan puheentunnistukseen, yksittäinen lause tulkitaan sarjaksi muuttujia, jotka ovat liittyneet toisiinsa tietyllä todennäköisyydellä kielen sanavaraston muodostaman tila-avaruuden sisällä. Matemaattisesti HMM-malli pyrkii ratkaisemaan maksimointiongelman, jossa pyritään tunnistamaan sanasarja vertailukohteina olevien akustisen mallin ja kielimallin avulla.

Syväoppimisella tarkoitetaan sellaisia koneoppimistekniikoita, joiden selkein ero perinteisiin koneoppimistekniikoihin on verkkokerrosten huomattavasti suurempi määrä.

Syväoppimisessa käytettäviä verkkoarkkitehtuureita ovat esim. syvät uskomusverkot, konvoluutiohermoverkot ja RNN- eli toistuvat hermoverkot. Vaikka syväoppimisessa voidaan käyttää sekä luokiteltua että luokittelematonta syöttödataa, sen vahvuuksia pystytään parhaiten hyödyntämään, kun syväoppimismallille syötetyn datan muuttujien tai parametrien määrä kasvaa hyvin suureksi. Tällöin luokittelemattoman datan mallintamiseen liittyvien virheiden määrä laskee niin pieneksi, että sen hyödyntäminen on kannattavaa. Toisin kuin perinteiset hermoverkot, syväoppimisverkot ovat vähemmän alttiita jäämään loukkuun tavoitefunktion paikallisiin optimeihin, mikä vähentää ylisovitusongelmia.

Syväoppimisen edut perinteisiin matemaattisiin malleihin verrattuna lähtevät siitä, kuinka syväoppimismallien rakenne pyrkii matkimaan ihmisen aivotoimintaa. Näissä malleissa alemman tason ominaisuudet voidaan erotella jo käytetyn verkon alemmista kerroksista ja verkon koko syvyys voidaan käyttää hyväksi korkeamman tason ominaisuuksien poimimiseen. Koneoppimisen viimeaikainen tutkimuskehitys on tapahtunut sellaisella alueella, joka on suosinut syväoppimista. Osoituksena tästä voidaan pitää nimettyjen syväoppimisarkkitehtuurien suurta määrää ja sitä, että uusien arkkitehtuurien julkistamisia odotetaan tapahtuvan säännöllisesti myös tulevaisuudessa. Puheentunnistuksessa käytettäviä malleja tutkittaessa on saatu todisteita perinteisten matemaattisten mallien ja syväoppimismallien ominaisuuksia yhdistävien ns. hybridimallien paremmuudesta tietyissä sovelluskohteissa. Näissä kohteissa hybridimallin suorituskyky oli parempi sanavirheiden määrässä mitattuna kuin perinteisen mallin tai syväoppimismallin suorituskyky yksin.

Lähteet

Dahl G., Yu D., Deng L. & Acero A. (2012). Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Transactions on Audio, Speech and Language Processing*, 30–42, 20(1). https://doi.org/10.1109/TASL.2011.2134090

Chen X. & Lin X. (2014). Big data deep learning: challenges and perspectives. *IEEE Access*, 2014(2), 514–525. https://doi.org/10.1109/ACCESS.2014.2325029

Davis D., Patronis E. & Brown P. (2013). *Sound System Engineering 4e*. 4. painos. Routledge. https://doi.org/10.4324/9780240818474

Hou J., Rabiner L. & Dusan S. (2006). Automatic speech attribute transcription (ASAT) – The front end processor. *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, Toulouse. https://doi.org/10.1109/ICASSP.2006.1660025

Laver J. (2012). *Principles of phonetics.* Cambridge University Press. https://doi.org/10.1017/CBO9781139166621

Nassif A., Shahin I., Attili I., Azzeh M. & Shaalan K. (2019). Speech recognition using deep neural networks: A systematic review. *IEEE Access 7,* 19143–19165. https://doi.org/10.1109/ACCESS.2019.2896880

Pouyanfar S., Sadiq S., Yan Y., Tian H., Tao Y., Reyes M., Shyu M., Chen S. & Iyengar S. (2019). A survey on deep learning: Algorithms, techniques, and applications. *ACM Computing Surveys, 51*(5). https://doi.org/10.1145/3234150

Pulkki V. (2015). *Communication acoustics: An introduction to speech, audio and psychoacoustics.* John Wiley & Sons, Incorporated. https://ebookcentral.proquest.com/lib/kutu/detail.action?pq-origsite=primo&docID=7104151

Rabiner L. & Juang B. (1986). An introduction to hidden Markov models. *IEEE ASSP Magazine, 3*(1), 4–16. https://doi.org/10.1109/MASSP.1986.1165342

Mitkov R. (2014). *The Oxford handbook of computational linguistics.* Oxford University Press. https://doi.org/10.1093/oxfordhb/9780199573691.001.0001

Reis A., Paulino D., Paredes H. & Barroso J. (2017). Using intelligent personal assistants to strengthen the elderlies' social bonds. Teoksessa M. Antona & C. Stephanidis (toim.), *Universal Access in Human–Computer Interaction. Human and Technological Environments.* (s. 593–602). Springer. https://doi.org/10.1007/978-3-319-58700-4_48

Van Houdt G., Mosquera C. & Nápoles G. (2020). A review on the long short-term memory model. *Artificial Intelligence Review, 53*(8), 5929–5955. https://doi.org/10.1007/s10462-020-09838-1

Song Z. (2020). English speech recognition based on deep learning with multiple features. *Computing, 102*(3), 663–682. https://doi.org/10.1007/s00607-019-00753-0

Kapargavalli S. & Chandra E. (2016). A review on automatic speech recognition architecture and approaches. *International Journal of Signal Processing, Image Processing and Pattern Recognition, 9*(4), 393–404. https://doi.org/10.14257/ijsip.2016.9.4.34

Jones M. & Knight R-A. (toim.) (2013). *The Bloomsbury Companion to Phonetics.* Bloomsbury Publishing plc. https://doi.org/10.5040/9781472541895

Kurimo M. (2009). Puheentunnistus. Teoksessa O. Aaltonen, R. Aulanko, A. Iivonen, A. Klippi & M. Vainio (toim.), *Puhuva ihminen* (s.336-343). Otava.

Bhagyaveni, M. A., Vishvaksenan, K. S., & Kalidoss, R. (2016). *Introduction to analog and digital communication* (1. painos). Denmark: River Publishers.

IBM & Microsoft (1991). *Multimedia Programming Interface and Data Specifications 1.0* Haettu 11.9.2024 osoitteesta https://docslib.org/doc/13165772/multimedia-programming-interface-and-data-specifications-1-0

Wang Y., Hou Y., Che W. & Liu T. (2020) From static to dynamic word representations: a survey. *International Journal of Machine Learning and Cybernetics, 11*, 1611–1630. https://doi.org/10.1007/s13042-020-01069-8

Gerard C. (2021). *Practical Machine Learning in JavaScript: TensorFlow.js for Web Developers* (s.1–8). https://doi.org/10.1007/978-1-4842-6418-8

Sakurai A. & Shinozawa Y. (2008). Linguistic productivity and recurrent neural networks. Teoksessa X. Hu & P. Balasubramaniam (toim.), *Recurrent Neural Networks.* (s. 43–60). INTECH. https://doi.org/10.5772/68

Deng L., Hinton G. & Kingsbury B. (2013). New types of deep neural network learning for speech recognition and related applications: an overview. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada,* 8599-8603. https://doi.org/10.1109/ICASSP.2013.6639344