Palauteoppimisen hyödyntäminen luonnollisen kielen tuottamisessa

Riku Vornanen

Kandidaatintutkielma
HELSINGIN YLIOPISTO
Tietojenkäsittelytieteen laitos

Helsinki, 27. huhtikuuta 2017

HELSINGIN YLIOPISTO — HELSINGFORS UNIVERSITET — UNIVERSITY OF HELSINKI

Tiedekunta — Fakultet — Faculty	Laitos — Institution — Department
Matemaattis-luonnontieteellinen	Tietojenkäsittelytieteen laitos

Tekijä — Författare — Author
Riku Vornanen

Työn nimi — Arbetets titel — Title

Palauteoppimisen hyödyntäminen luonnollisen kielen tuottamisessa

Oppiaine — Läroämne — Subject
Tietojenkäsittelytiede

Työn laji — Arbetets art — Level	Aika — Datum — Month and year	Sivumäärä — Sidoantal — Number of pages
Kandidaatintutkielma	27. huhtikuuta 2017	18

Tiivistelmä — Referat — Abstract

Tiivistelmä.

Avainsanat — Nyckelord — Keywords
palauteoppiminen, luonnollisen kielen tuottaminen, hierarkkinen palauteoppiminen

Säilytyspaikka — Förvaringsställe — Where deposited

Muita tietoja — Övriga uppgifter — Additional information

Sisältö

1 Johdanto

Tekniikan ja tekoälyn kehitys on ollut nopeaa viimeisten vuosien aikana. Kone on ohittanut ihmisen jo monella alalla ja tullut etevämmäksi monessa tarkkuutta ja laskentatehoa vaativassa tehtävässä. On kuitenkin useita tehtäviä, joista keinoäly ei pysty niiden monimutkaisuuden ja vaikean määriteltävyyden vuoksi vielä selviytymään.

Yksi tällainen tehtävä on luonnollisen kielen tuottaminen. Ihmisten käyttämään luonnolliseen kieleen liittyy paljon erilaisia mekanismeja, joilla ilmaistaan merkityksiä, eikä niitä ole helppo mallintaa koneen ymmärtämään muotoon. Tässä työssä esitellään tietotekninen keino nimeltä palauteoppiminen, jonka avulla kone voi oppia kielen toimintaa.

Tämä työ käsittelee siis palauteoppimista (engl. reinforcement learning) ja sen hyödyntämistä luonnollisen kielen tuottamisessa. Työn tarkoituksena on osoittaa, että luonnollisen kielen kokonaisvaltainen oppiminen on tekoälysovellukselle suuri haaste, ja vakuuttaa lukija siitä, että palauteoppiminen on erittäin käyttökelpoinen ratkaisu näiden vaikeuksien vähentämiseksi. Aiheen käsittelyssä tuodaan esille palauteoppimisen toiminnan yhteneväisyyttä ihmisaivojen tapaan oppia kieltä, sillä ihminen on toistaiseksi ainoa entiteetti, joka tätä itse kehittämäänsä ilmaisujärjestelmää on onnistunut tehokkaasti oppimaan ja käyttämään.

Ensimmäiseksi esitellään luonnollisen kielen tuottamiseen liittyviä haasteita ja perehdytetään pintapuolisesti, mutta keskeisten ongelmien kannalta riittävällä tasolla, luonnollisen kielen monitasoisiin rakenteisiin. Tarkoitus on osoittaa, että luontevan kielenkäytön matkiminen ei ole yksinkertaista ja sitä vaikeuttavat niin erilaiset rakenteet, kuin tilannesidonnaisuuskin.

Tämän lisäksi määritellään mitä palauteoppimisella tarkoitetaan ja esitellään koneoppimisen käsite. Palauteoppimisen periaate selitetään ensin sen ihmisaivojen oppimistapaan vertautuvan, kokeiluun ja palautteeseen perustuvan mekanismin kautta. Tämän jälkeen esitellään, miten teoriaa on lähdetty toteuttamaan tietojenkäsittelytieteen keinoin. Tähän liittyen käydään läpi matemaattinen malli, Markovin päätöksentekoprosessi, jota käytetään kuvaamaan palauteoppimisen toimintamekanismia ja ympäristön kanssa vuorovaikutuksessa olemista. Samassa esitellään, miten prosessin avulla voidaan mallintaa keskustelu.

Työssä tarkoitus on esitellä, miten palauteoppimisen keinoilla on mahdollista opetella erilaisia kielen tuottamiseen liittyviä tehtäviä. Työssä annetaan asiasta esimerkkejä ja lopussa esitellään vielä palauteoppimista hyödyntäviä luonnollista kieltä tuottavia järjestelmiä.

2 Luonnollisen kielen tuottamisen haasteet

Luonnollisen kielen tuottaminen on yksi tekoälysovellusten ja käyttöliittymäsuunnittelun ajankohtaisia haasteita. Oikeaoppista kieltä on helppo tuottaa, mutta dialogin käyminen ihmisen kanssa vaatii kielen tulkintaa ja useiden muuttujien huomioon ottamista. Kielen tulkitsemisen lisäksi pitäisi myös osata muodostaa luonteva ilmaisu, joka sopii keskustelun kulkuun ja tilanteeseen tuoden samalla halutun viestin perille tarkoituksenmukaisella tavalla.

Ensimmäisessä luvussa perehdytään luonnollisen kielen tutkimukseen ja tietokonelingvistiikan rooliin siinä. Lisäksi käsitellään luonnollisen kielen monitasoisuutta ja monimutkaisuutta. Kielen roolia ihmisen maailmassa ja toiminnassa havainnollistetaan esittelemällä Ludwig Wittgensteinin kielipelin käsite.

2.1 Luonnollisen kielen tutkimus

Luonnollista kieltä tutkitaan useilla eri tieteenaloilla. Kielitieteilijät eli lingvistit tutkivat esimerkiksi, miten sanat muodostavat lauseita ja miksi toisilla peräkkäisistä sanoista koostetuilla lauseilla on merkitys ja toisilla ei. Lingvistit käyttävät apunaan muun muassa matemaattisia malleja ja analysoivat kielen rakennetta. Psykolingvistit taas tutkivat, miten ihmiset ymmärtävät kielen rakenteita ja sanojen merkityksiä. Tutkimuksissa mitataan, miten ihmiset itse käyttävät kieltä ja ymmärtävät toisiaan. Filosofit taas tutkivat esimerkiksi kielellisten ilmauksien merkityksen olemusta ja sitä, miten sanat selittävät ympäröivää maailmaa [JaA95].

Nämä eri näkökulmat kielentutkimukseen kertovat jo siitä, että aihe ei ole yksinkertainen ja että kielen ymmärtämiseen liittyy useita puolia. Tietokonelingvistiikan tavoitteena on kehittää teoria kielestä tietojenkäsittelytieteen välineiden, kuten algoritmien ja tietorakenteiden sekä tekoälyn tekniikoiden avulla. Sen on väistämättä hyödynnettävä kehitystyössä näiden muiden tieteenalojen tuottamaa tietoa. Tietokonelingvistiikan tyypillisiä ongelmia ovat esimerkiksi, miten lauseiden rakenne tunnistetaan, miten ymmärrys ja järkeily voidaan mallintaa ja miten kieltä voi käyttää tiettyjen tehtävien suorittamisessa [JaA95].

2.2 Luonnollisen kielen eri tasot

Kuten edellä mainituista tietokonelingvistiikan kohtaamista haasteista käy ilmi, ihmisten tapa käyttää kieltä ei ole mallinnettavissa yksiselitteisesti kaavoiksi. Vaikka ihmisten kehittämän kielen sääntöjä kootaan yhteen ja kirjakieltä opetetaan koulussa, on kieliopin ja sanojen virallisten merkitysten avulla tuotetuilla sanajonoilla ilmaisuvoimalliset rajoituksensa. Kulttuuri, fyysinen ympäristö ja keskustelijoiden aikaisempi kanssakäyminen vaikuttavat merkityksiin ja viittauksiin, joita valitut sanat tuovat mukanaan. Lisäksi

nämä taustavaikuttajat ovat itse jatkuvassa muutoksessa.

Asiayhteys, jossa lausetta käytetään, saattaa muuttaa yksinkertaisen lauseen merkityksen täysin tai tuoda siihen jonkun uuden kulman. Ihmiselle intuitiivisesti ilmiselvä tulkinta jostakin lauseesta saattaa koneelle olla hyvin vaikea saavuttaa. Esimerkiksi suomenkielinen lause "Käytkö usein täällä?" on kirjakielen sääntöjen ja sanojen virallisten merkitysten perusteella kysymys, jolla halutaan saada tietää käykö kysymyksen vastaanottaja usein paikassa, jossa kysymys esitetään. Olemme kuitenkin sitoneet sananparteen kulttuurillisia merkityksiä ja, jos ilmausta käytetään esimerkiksi ravintolassa pikkutunneilla, osaamme olettaa, että kyse on todennäköisesti jonkinlaisesta pariutumisriitteihin liittyvästä vihjailusta. Tällaisia merkityksiä ymmärtääkseen on ihmisenkin opittava niihin liittyvä taustatieto.

Ihminen kerää elämänsä aikana paljon tietoa ympäröivästä maailmasta ja asiayhteyksistä. Tämä taustatieto on kielen osaamisen lisäksi vaatimus sille, että yhteenkään keskusteluun on mahdollista mielekkäästi osallistua. Luonnollista kieltä ymmärtääkseen on oltava tietoa kielen eri tasoista. Foneetiikka liittyy ääneen puhutun kielen tulkitsemiseen ja kertoo, miten sanat liittyvät niitä kuvaaviin ääniin. Morfologinen tieto on tietoa siitä, miten sanoja johdetaan perusmuodostaan esimerkiksi päätteiden ja taivutusten avulla. Syntaktisen tiedon mukaan kootaan sanoja lauseiksi ja määritellään, mikä kunkin sanan rooli on lauseen rakenteen kannalta. Semantiikka taas ottaa kantaa sanojen määritelmälliseen merkitykseen ja sanoista johdettujen lauseiden merkitykseen. Semanttinen merkitys on kontekstista riippumaton eikä ota juurikaan huomioon kielen muita tasoja. Pragmaattinen eli käytännöllinen tieto on tietoa sanojen käytöstä eri tilanteessa ja miten niiden käyttö vaikuttaa lauseen tulkintaan. Keskustelun onnistumiseksi myös oltava keskustelujen normeihin liittyvää tietoa ja tiedettävä peräkkäisten ilmausten vaikutuksesta toisiinsa. Lisäksi on oltava yleistä tietoa maailmasta ja esimerkiksi toisten ihmisten uskomuksista ja tavoitteista [JaA95].

2.3 Kielipelit

Yksi 1900-luvun alun kuuluisimmista filosofeista, perinteistä filosofiaa kielen olemuksen ymmärtämättömyydestä kritisoinut itävaltalainen Ludwig Wittgenstein kehitti uransa loppupuolella niin kutsutun kielipelin käsitteen. Kielipeli kuvaa ihmisten maailmaan sidoksissa olevaa tapaa käyttää kieltä havainnollistaen hyvin sanojen ja lauseiden merkitysten eksaktin määrittelyn mahdottomuutta.

Kielipelit tarkoittavat tiettyjä aktiviteetteja, joissa kieli on pääosassa [ArM11]. Erilaisia kielipelejä ovat esimerkiksi kiittämisen, käskemisen, rukoilemisen, tervehtimisen ja vitsailun kielipelit. Kieli ei ole erillinen ilmiö, vaan kytkeytyy ihmisen muuhun toimintaan. Merkitystä voi eritellä vain eri käyttöyhteyksien valossa [SaE99]. Wittgensteinin mukaan kielen osaaminen on erilaisten ilmausten käyttämistä niille kuuluvissa kielipeleissä [ArM11].

Kuten todettua, edellä mainittu esimerkkilause “Käytkö usein täällä?” ei ole yleensä tulkittavissa sananmukaisesti, vaan saa merkityksensä siitä, miten sitä on ollu tapana käyttää. Se on siis esimerkki eräänlaisesta kielipelistä. Teoksessaan filosofisia tutkimuksia (Saks. philosophische untersuchungen) Wittgenstein käyttää yhtenä kielipelejä selventävänä esimerkkinä talonrakentajia. Rakentaja A on katolla kokoamassa taloa erimuotoisten kivien avulla. Kivet on lajiteltu luokkiin: kuutio, pylväs, laatta ja palkki. Rakentaja B:n on ojennettava kiviä A:n tarvitsemassa järjestyksessä. Tarkoitusta varten rakentajat käyttävät kieltä, johon kuuluvat sanat “kuutio”, “pylväs”, “laatta” ja “palkki”. A huutaa sanan ja B tuo kiven, jonka on oppinut tuomaan käskyn kuullessaan. Rakentajien kielipeli havainnollistaa, miten sanojen merkitys syntyy niiden käyttötarkoituksesta. On helppoa kuvitella tilanne, jossa huudettaessa sana “kuutio”, on ilmaisun merkitys toisen kielipelin ollessa kyseessä täysin toinen.

2.4 Lauserakenteet ja tulkinta

Luonnollinen kieli on monimutkainen kokonaisuus ja kaikki ilmaisut tuntuvat olevan erillisiä “käyttötapauksia”, joissa on otettava huomioon suuri määrä taustavaikuttajia. Kuitenkin kielen rakennetta ja lauseita voi analysoida luokittelemalla sanoja niiden käyttötarkoituksen mukaan. Esimerkiksi substantiiviksi luokiteltu sana toimii todennäköisesti lauseen objektina tai subjektina ja verbi predikaattina. Näiden roolien päättelyssä auttaa syntaktinen tieto ja sanojen tarkoituksen tunnistamisessa esimerkiksi semanttinen-, pragmaattinen- ja morfologinen tieto. Luonnollista kieltä tuotettaessa näitä sääntöjä tarvitaan, jotta oppiminen ei olisi pelkkää raskasta eri tapausten ulkoa opettelua. Yleistäminen ja luokittelu ovat kielen tulkinnassa välttämätön väline, sillä ne mahdollistavat kommunikaation kokoamalla kielen käyttäjien sanoille ja lauserakenteille yhteisesti tunnistamat merkitykset.

Kuva 2.1 Lauseen syntaktisen rakenteen jäsennyspuu

Kuva 2.1 havainnollistaa, miten sama lause voidaan jäsentää kahdella eri tavalla. Ensimmäinen lause tarkoittaa, että lauseen käyttäjä on nopeampi uimaan, kuin eläin koira. Toinen lause taas sitä, että puhuja ui koiraksi nimitetyllä uintitekniikalla nopeammin, kuin jollakin toisella tekniikalla. Syntaktisen tiedon perusteella osataan sanoa, että on olemassa ainakin nämä kaksi vaihtoehtoa. Jotta voidaan päätellä kumpi tulkinta on oikea, on oltava tietoa kontekstista ja kontekstin vaikutuksesta tulkintojen todennäköisyyksiin. Jos edellisessä lauseessa on puhuttu eri uintitekniikoista, vaikuttaa toinen vaihtoehto todennäköiseltä. Jos taas aikaisemmin on puhuttu esimerkiksi eläinten uintinopeuksista, pitäisi osata päätellä, että kyseessä on mitä luultavimmin ensimmäinen vaihtoehto. Tällaiset intuitiiviset päätelmät voidaan oppia hyödyntämällä heuristisia, eli kokeilevia ja arvailevia prosesseja, kuten esimerkiksi yrityksen ja erehdyksen periaatetta.

3 Palauteoppiminen

Kieltä tulkitsevan algoritmin on osattava ottaa huomioon edellä mainitut esimerkit luonnollisen kielen eri tasoista. Kielen tuottaminen taas asettaa sitä enemmän haasteita, mitä monipuolisempaa kieltä halutaan tuottaa. On vaikea kirjoittaa näitä kaikkia kulmia huomioon ottavaa, tarkkoihin sääntöihin perustuvaa koodia, minkä vuoksi kokemuspohjainen koneoppiminen ja varsinkin sen alalaji, palauteoppiminen, on käytännöllinen väline pulman käsittelyssä. “Sanan merkitys on sen käyttö kielessä” Wittgenstein on kiteyttänyt. Sanat eivät siis ole nimilappuja tietyille asioille, vaan niillä on monia käyttötarkoituksia. Palauteoppimista voidaan käyttää keinona opetella näitä käyttötarkoituksia ja kielipelejä eri yhteyksissä.

Palauteoppiminen on koneoppimisen haara, jossa oppiva järjestelmä kommunikoi ulkoisen ympäristön kanssa ja kehittää toimintaansa siitä saamansa palautteen perusteella. Sitä on kutsuttu pienen mittakaavan tekoälyongelmaksi, koska suoriutuakseen tehtävistään hyvin ja tehokkaasti, on palauteoppimista toteuttavan järjestelmän kyettävä toimimaan autonomisesti [LiM15].

Teknologian myötä saatavilla olevan tiedon määrän lisääntyminen viime vuosina on luonut tarvetta tehokkaampien tietojenkäsittelymenetelmien kehittämiseen eri aloilla. Tämä kehitys on osaltaan johtanut palauteoppimisen teorian ja käytännön edistymiseen mahdollistaen sen entistä monipuolisemman hyödyntämisen konkreettisen maailman ongelmissa [LiM15].

Alla käydään läpi, mitä koneoppiminen yleisesti on ja minkälaisissa tilanteissa sitä käytetään. Sen jälkeen syvennytään palauteoppimiseen, sen periaatteeseen ja siihen liittyvään Markovin päätöksentekoprosessiin. Lopuksi avataan palauteoppivan järjestelmän keskustelun mallintamisen periaate ja selitetään miten kielen esitys tuotetaan käytännössä (engl. surface realization).

3.1 Koneoppiminen

Koneoppiminen on tietokoneiden ohjelmoimista optimoimaan toimintaansa esimerkkiaineistoa tai kerättyä kokemusta hyödyntäen [AlE04]. Tehokkainta on tietenkin hyödyntää molempia lähestymistapoja. Esimerkkiaineiston perusteella saa hyvän osaamispohjan aikaiseksi, eikä koneoppivan järjestelmän tarvitse opetella kaikkia opeteltavan asian säännönmukaisuuksia keräämällä niitä kokeilemalla. Luonnollisen kielen oppimisen tapauksessa on tehokasta käyttää aluksi laajasta kieliaineistosta kerättyä tietoa ja siltä pohjalta kehitetystä toiminnasta saadun palautteen perusteella kehittää myöhemmin ilmaisuja ja lisätä sääntöjä asioiden välisistä yhteyksistä. Ilman tätä pohjaa on järjestelmän lähdettävä keräämään tietoa tyhjästä ja oppiminen kestää kauemmin.

Tietokoneita on niiden historian alusta asti käytetty suorittamaan lasken-

taa, jonka ihminen osaa ohjeistaa, mutta johon ihmisen aivojen kapasiteetti ei riitä. On tiedetty, mitä halutaan laskea ja kirjoitettu ohjelma, joka hyödyntää koneen ihmiseen nähden ylivoimaista laskutehoa. Koneoppimista tarvitaan tapauksissa, joissa ongelman ratkaisun laskemista varten ei voi suoraan kirjoittaa ohjelmaa.

Oppiminen on pakollista ohjelman työskennellessä sellaisen aiheen parissa, josta ihmisellä ei ole tarpeeksi tietoa tai johon liittyvä tieto on vaikea selittää. Esimerkiksi ääneen puhutun puheen tunnistaminen on meille ihmisille helppoa, mutta emme osaa selittää, miten teemme sen. Erilaiset ihmiset ääntävät sanat eri tavoin iästä, sukupuolesta ja aksentista riippuen. Koneoppimisessa asia yritetään ratkaista keräämällä iso määrä esimerkkiaineistoa eri ihmisten ilmaisuista ja opetellaan yhdistämään ne sanoihin [AlE04].

Koneoppimista toteuttavaa itsenäistä järjestelmän osaa kutsutaan oppimisagentiksi (engl. learning agent). Oppimisagentti kerää ympäristöstä tietoa ja laskee näiden tietojen sekä koneoppimista toteuttavien kaavojen perusteella todennäköisesti tilanteeseen nähden parhaan tavan toimia.

Koneoppiminen on yläkäsite kaikille jonkinlaista oppimista sisältäville toiminnoille, joita tietokone suorittaa. Näiden toimintojen pohjana on yleensä esimerkiksi todennäköisyyksien perusteella ennustaminen ja tarkasteltavien asioiden luokittelu.

Sääntöpohjaisessa koneoppimisessa todennäköisyyksien perustana ovat usein assosiaatiosäännöt. Niitä hyödynnetään esimerkiksi ostoskorianalyyseissä. P(Y/X)=0.6 tarkoittaa 60

Yksi koneoppimisen väline on luokittelu. Siinä asiat luokitellaan kategorioihin loogisten sääntöjen, esimerkiksi jos-niin -tyyppisten lauseiden perusteella [AlE04]. Luokitteluun voi myös yhdistää todennäköisyyksiä, jotta luokittelu ei jää binääriseksi 0/1 -arvioinniksi. Luokittelua käytetään esimerkiksi kirjoitetun tekstin kirjainten tunnistamiseen visuaalisesti. Luokkia on yhtä monta kuin kirjaimia ja kirjaimet opitaan sijoittamaan oikeisiin luokkiinsa arvioimalla niitä joko tiettyjen ulkonäkösääntöjen perusteella tai vertaamalla niitä luokkaan jo valmiiksi sijoitettujen kirjainten kuviin.

Näiden luokittelun ja assosiaatiosääntöjen kaltaisten välineiden käyttämistä aineiston luokittelussa kutsutaan ohjatuksi oppimiseksi (engl. supervised learning). Juuri tällaisten kirjoitettuihin sääntöihin perustuvien keinojen käyttö osoittautuisi luonnollisen kielen tuottamista vaativissa tehtävissä tehottomaksi, kappaleessa 2 kuvattujen kielen moniulotteisuuksien vuoksi. Ohjaamaton oppiminen (engl. unsupervised learning) eroaa ohjatusta oppimisesta siten, että siinä ei luokkia ja sääntöjä ole valmiiksi, vaan se hyödyntää pelkästään syötteenä saamansa tietoa ja jakaa aineiston itse luokkiin. Ohjaamattomassa oppimisessa pyritään usein tunnistamaan tietoaineistosta rakenteita ja toistuvia kuvioita.

Palauteoppimista on vaikea sijoittaa tässä jaottelussa näihin kumpaankaan oppimisen kategoriaan, koska sen tehtävät ja tavoite ovat erilaisia. Kun ohjaamattomassa oppimisessa etsitään eroja ja samankaltaisuuksia tie-

toaineistosta, palauteoppimisessa pyritään löytämään paras toimintamalli kaikkiin tilanteisiin lopullisen palkinnon maksimoimiseksi. Ohjaamattomassa oppimisessa toiminnasta ei saada palautetta tai palkintoa, vaan se on enemmänkin aineiston käsittelyyn tarkoitettu väline. Ohjatussa oppimisessa taas hyödynnetään selkeää palautetta, mutta siinä tiedot yhdistetään luokkiin ajattelematta pitkän aikavälin palkitsevuutta.

3.2 Palauteoppimisen periaate

Tyypillisesti oppimiseen ajatellaan liittyvän kanssakäyminen ympäristön kanssa. Ihminen on aistiensa välityksellä yhteydessä ympäristöönsä ja kerää tietoa asioiden yhteyksistä muodostaen toimintamalleja, joiden avulla saavuttaa tavoitteitaan. Ilman kokemuksen ja tiedon kartuttamista on vaikea kuvitella minkäänlaista oppimista tapahtuvan. Tämä ympäristön kanssa vuorovaikutuksessa oleminen on melkein kaikkien oppimista ja älykkyyttä koskevien teorioiden pohjalla [SuB12].

Palauteoppiminen eli vahvistusoppiminen (engl. reinforcement learning) on koneoppimisen yksi lähestymistapa, joka muistuttaa ihmisen tapaa oppia toimimaan elinympäristössään tekojensa seurauksien perusteella. Mekanismiin liittyy olennaisesti yrityksen ja erehdyksen periaate. Toiminta, joka tuottaa negatiivisen lopputuloksen korvataan ensi kerralla toisenlaisella toiminnalla ja positiiviseen lopputulokseen johtavia valintoja suositaan. Päätöksiä tehdessä käytetään aina toimintamallia, joka olemassa olevan tiedon perusteella tuottaa parhaan mahdollisen lopputuloksen.

Koneellisen palauteoppimisen periaate on yhdistellä eri tilanteita ja toimintoja tuottaen mahdollisimman korkean numeerisen palkinnon. Koneoppimiseen liittyy yleensä annettujen tai ohjelmoitujen ohjeiden mukaan toimiminen, mutta palauteoppimisen tapauksessa oppiminen tapahtuu yrityksen ja siitä saadun palautteen avulla. Vain mekanismi, jolla toiminnasta saatu palaute vaikuttaa käyttäytymiseen on ohjelmoitu valmiiksi [SuB12].

Monimutkaisemmissa tapauksissa on mahdollista, että palauteoppimista käyttävän agentin on otettava huomioon toiminnastaan suoraan aiheutuvan palkinnon lisäksi pidemmän aikavälin palkitsevuus [SuB12]. Tällöin on laskettava toiminnon tuottaman tilan mahdollistamien toimintojen potentiaaliset palkinnot ja taas edelleen niiden toimintojen tuomien tilojen toimintojen palkinnot. Tätä ketjua on jatkettava mahdollisimman pitkään arvioiden, mikä yhdistelmä tuottaa lopputulokseksi suurimman numeeristen palkintojen summan.

Optimoidakseen toimintaansa palauteoppimista toteuttavan agentin on kyettävä hyödyntämään opittua tietoaan käyttämällä tehokkaiksi todettuja toimintoja, mutta myös etsittävä parempia vaihtoehtoja käyttämällä ennen kokeilemattomia toimintoja. Ongelma syntyy siitä, että kumpaakaan vaihtoehtoa ei voi täysin toteuttaa samanaikaisesti sulkematta toista pois. Tämä olemassa olevan tiedon hyödyntämisen ja uuden tiedon tutkimisen välillä

tasapainottelu (exploitation-exploration trade-off) on keskeinen palauteoppimiseen liittyvä haaste [SuB12].

Tämä tasapainottelu vertautuu ajatukseen siitä, että kriittistä ajattelua harjoittavan ihmisen, esimerkiksi tieteen tekijän, on järkevästi toimiakseen käytettävä totena pidettyjä oletuksia maailmasta mutta samalla oppiakseen ja kehittyäkseen oltava valmiina vastaanottamaan uutta tietoa. Kun tarpeeksi todisteita vanhan teorian kumoavan tiedon puolesta ilmenee, on kyettävä kyseenalaistamaan vanha tieto ja tarvittaessa vaihdettava käsitystään asiasta.

3.3 Markovin päätöksentekoprosessi

Palauteoppimista kuvaillaan oppijan ja palautetta antavan ympäristön vuorovaikutukseksi. Ympäristön käsitteellistämisessä käytetään usein matemaattista mallia nimeltä Markovin päätöksentekoprosessi (Markov decision process/MDP). Mallia hyödynnetään tilanteissa, joissa päätöksen tekemiseen vaikuttavat tekijät ovat osittain satunnaisia tai niistä on vaikeaa saada tarpeeksi tietoa.

Markovin päätöksentekoprosessin yksinkertaisin versio voidaan mallintaa monikkona S, A, T, R [SuB12].

S = s, s, s...s on äärellinen tilatietojen joukko, jossa n on jokin kokonaisluku. Se sisältää esimerkiksi toimintaympäristön muuttujien tämän hetkisen tilan ja tiedon seuraavaksi suoritettavasta toiminnasta. Agentin on päästävä tarkkailemaan tilojen toimintoja ja vaikutuksia, jotta se voi käyttää niitä oppimiseen. Kun toiminnon a suorittaa tilassa s aika-askeleen t kohdalla, kertoo uusi tila s'=st + 1 toiminnan vaikutuksen ympäristöön.

A = a, a, a..., a on äärellinen toimintojen joukko, jossa m on jokin kokonaisluku. Se on lista mahdollisista yksiselitteisistä toiminnoista, kuten vaikka tiettyjen sanallisten ilmausten käyttäminen. Määritellyt toimintavaihtoehdot antavat pohjan yrityksen ja erehdyksen periaatteella toimimiselle.

T on todennäköisyyteen perustuva tilanvaihdosfunktio. Sen funktio P(s|s, a) kertoo mihin tilaan s' toiminto a tilassa s johtaa ja millä todennäköisyydellä.

R on palkintofunktio R(s|s, a), joka kertoo minkä numeerisen palkinnon toiminta a tilassa s tuottaa.

Kun kanssakäynti prosessia käyttävän oppimisagentin ja ympäristön kanssa alkaa, agentti saa ympäristöltä tilatiedon s 0 S ja suorittaa toiminnon a 0 A. Kuvassa 3.1 on kuvattuna prosessi, jossa agentti saa kuvauksen sen hetkisestä tilasta s ja palkinnosta r. Sen täytyy suorittaa toiminta a A ja palautteena se saa palkinnon r R, minkä lisäksi se saa tietoonsa seuraavan tilan s S.

Prosessi muodostaa rajallisen listan tiloja, toimintoja ja palkintoja: s, a, r, s, a, . . ., r*, s*. Merkit r* ja s* kuvaavat viimeisintä palkintoa ja viimeistä tilaa kanssakäymisen tultua päätökseensä. Tilojen ja toimintojen yhdistämisiä toisiinsa kutsutaan käytännöiksi (engl. policy). Agentin tehtävä on löytää

optimaalinen käytäntö, jossa kaikista tiloista s on tiedossa sopiva toiminta a, niin että käytännön lopputuloksena on odotettavissa mahdollisimman suuri palkinto [DeN15].

Eri tilojen painoarvoja säädellään symbolilla kuvattavalla muuttujalla, jonka arvo on 0 ja 1 välillä. Arvon ollessa 0 otetaan huomioon vain välitön palkinto ja sen lähestyessä arvoa 1, agentti ottaa huomioon pidemmän aikavälin palkinnon. Esimerkiksi päättymättömässä loputon horisontti -mallissa (engl. infinite horizon), se asetetaan vähentämään tulevien palkintojen merkitystä. Palkinnot summautuvat tällöin: r+ r+ r. . . , alaindeksin t merkitessä, aina odotettavien palkintojen määrän kasvaessa yhdellä kasvavaa, aika-askelta.

Kuva 3.1: Markovin mallin mukainen agentin ja ympäristön kanssakäyminen

Markovin päätöksentekoprosessista on monta johdosta ja variaatiota - yllä esitelty on vain sen perustoiminta. Prosessi pohjautuu palauteoppimisen periaatteeseen ja toimii perusmekanismina monille palauteoppimista hyödyntäville koneoppimisen malleille.

3.4 Keskustelun mallintaminen Markovin prosessin avulla

Keskustelu voidaan mallintaa palauteoppimista varten käyttämällä Markovin päätöksentekoprosessin kaavaa. Keskustelun käytäntöjen oppija voidaan kuvata agenttina, joka kulkee toistensa kanssa yhteydessä olevien keskustelutilojen muodostaman verkon läpi [RiO11].

Kuva 3.2 keskustelun mallinnus MDP:n avulla [ScJ06].

Markovin päätöksentekoprosessin tilatietojen joukko S on yleensä rajallinen joukko ennalta määrättyjä tilamuuttujia. Nämä tilat voivat esittää keskusteluhistorian tuottamaa dialogin sen hetkistä tilannetta. Monien systeemien tilat ovat moniparametrisia ja sisältävät esimerkiksi keskusteluhistorian kuvauksen, dialogin toisen osapuolen eli yleensä ohjelman käyttäjän edellisen puheenvuoron piirteet ja käyttäjän odotetun tavoitteen [RiO11].

Kuvassa 3.2 on kuvattuna yksinkertaisen lentojenvarausjärjestelmän Markovin päätöksentekoprosessiin perustuva verkko dialogin tuottamista varten. Tilat sisältävät keskustelun kannalta oleellisia tietoja tallettavia muuttujia, jotka kuvaavat keskustelun sen hetkistä tilaa. Keskustelun tilaa kuvaava informaatio esitetään muuttujilla, jotka ovat kuvassa $\mathrm{orig}_city, dest_city, depart_date ja depart_time. Tilojenmal$

Toimintojen joukko A sisältää kaikki agentin käytettävissä olevat toimintavaihtoehdot. Toiminnot on keskustelun mallinnuksessa yleensä rajattu pieneen määrään toimintoja, kuten ”pyydä tietoja”, ”vahvista” tai ”esitä tietoja”. Hieman monipuolisempi vaihtoehto on esittää toiminto joukkona, jossa on muuttujia. Muuttujia voivat olla esimerkiksi puhetoiminto, muuttujan nimi tai muuttujan arvo. Kuvassa 3.2 on kuvattuna keskustelutoiminto a, jossa vahvistetaan keskustelukumppanille toiminnolla $\mathrm{impl}_conf, että kohde kaupunki on ”london” ja tiedustellaan requ_info - toiminnolla lähtöpäivää. Yleensä$
$Requ_info, depart_date > tuottamailmaisuvoisikuuluasuomeksi”Minäpäivänähaluatte lähteä?”tai, yk$

ɹ*Annalähtöpäivä*.ɹ.

Tilanvaihdosfunktio T kertoo, mikä olemassa olevassa tilassa suoritettu toiminto johtaa todennäköisesti mihinkin tilaan. Kuvassa 3.2 visuaalisesti mallinnetussa verkon osassa tilassa s suoritettu toiminto a johtaa tilaan s väistämättä eli todennäköisyydellä 1. Tilassa s on esimerkiksi tiedusteltu lähtöpäivää, jolloin kysymykseen saadaan vastaus ja depart$_d$*atemuuttuuarvoon*ɹ*known*ɹ.*Samallakaupunkivar*
*nollessatilassa*ɹ*confirmed*ɹ.*Toimintoataasjohtaa*, *jokotilaanstais*, *joistamolemmillasiirtymilläon*
ajan.Samallavarmistetaanjokatapauksessakohdekaupunki.Tämäjohtaapalauteoppimisenavullas

Palkintofunktio R taas kertoo siirtymien tavoittamien tilojen tuottamat palkinnot. Palkinnon voi olla määritelty kasvavan, kun informaatiota käyttäjän tarpeesta saadaan. Koska esimerkin tilaan s siirtyminen tuottaa enemmän informaatiota ja suuremman palkinnon kuin tilaan s siirtyminen, kannattaa valita todennäköisemmin isomman palkinnon, tilassa s suoritettuna, tuottama toiminto a. Keskustelustrategiaa varten palkintojen painoarvoa säätelevä muuttuja asetetaan yleensä lähelle arvoa 1, jotta palkinnot otettaisiin huomioon mahdollisimman pitkällä tähtäimellä. Jos kuvan tilojen jälkeen toiminnoilla voisi siirtyä vielä uusiin tiloihin saattaisivat niiden tuottamat potentiaaliset palkinnot muuttaa toimintojen pitkän aikavälin palkitsevuutta.

Samanlaisella mallinnuksella voidaan tuottaa myös enemmän tilamuuttujia ja toimintavaihtoehtoja sisältävää dialogia. Erilaisten tilojen ja toimintojen tarve kasvaa kuitenkin nopeasti, joten tehokkainta olisi, mikäli tällaiset tiettyihin tilanteisiin ja kielipeleihin erikoistuneet agentit voisivat toimia yhteistyössä, mutta kukin omasta osaamisalueestaan huolehtien. Seuraavassa luvussa käsiteltävä hierarkkinen palauteoppiminen on tähän ajatukseen perustuva tekniikka.

4 Palauteoppivat järjestelmät

4.1

4.2 Esimerkki kielen syntaksin opettelusta

Robotics and Autonomous Systems -julkaisun artikkelissa [MaD15] tutkittiin robottijoukon keskinäistä kielenkäyttöä, niiden pyrkiessä kuvailemaan testiympäristöään toisilleen. Tutkimus demonstroi hyvin palauteoppimisen hyödyntämistä kielen opettelussa. Testiympäristö oli yksinkertainen ja siinä käytettiin neljää esinettä (kirja, kynä, silmälasit ja pallo) ja kahta suhdetta (vasemmalla ja oikealla). Roboteilla oli yhteinen sanasto näille muuttujille ja oma kehittyvä kielioppinsa. Ne mukauttivat testin aikana palauteoppimisalgoritmin avulla kielioppinsa yhtenäisiksi kokeilemalla eri sanajärjestyksiä ja pystyivät lopulta kuvailemaan ympäristöä toisillensa yhteisellä kielellä. Toisin sanoen ne kehittivät tarkoitukseen soveltuvan kielipelin.

Lauseen sanajärjestyksille muodostui testin alussa kolme vaihtoehtoa: <Lause> = <esine><suhde><esine> p, <esine><esine><suhde>p, <suh-

de><esine><esine>p.

Kaavassa näkyvät kommunikointitilanteessa käytettävät syntaktiset vaihtoehdot. Todennäköisyydet p, p ja p alustettiin kaikki aluksi arvoon . Robotit kuvailivat pareina toisilleen ympäristöä ja kun ne käyttivät samaa sanajärjestystä samasta tilanteesta, palauteoppimisalgoritmi sai positiivisen palkinnon ja kasvatti tapauksen todennäköisyysfunktion p arvoa. Yhteinen kielioppi tulkittiin saavutetuksi, kun kaikki robotit käyttivät samoja ilmaisuja samoista tilanteista. Aikaisemmin mainitun hyödyntämisen ja tutkimisen välillä tasapainottelun järkeväksi toteuttamiseksi käytettiin satunnaisuuteen perustuvaa päätösmallia, pseudokoodina:

If [random] then
[satunnaisgeneroitu valinta]
else
[valitse ilmaisu, jonka todennäköisyys on suurin]
end if.

Testiä toistettiin myös asetelmassa, jossa yksi roboteista oli varustettu oikealla ihmisten kieliopilla, eikä sen ei tarvinnut itse opetella mitään. Muiden robottien palauteoppimisjärjestelmä palkitsi niiden käyttämän ilmaisun, jos se oli samankaltainen tämän “ihmisrobotin” käyttämän ilmaisun kanssa. Oikealla kieliopilla varustetun robotin käyttö nopeutti yhteisen kieliopin löytämistä varsinkin silloin, kun robotteja oli joukkueessa enemmän. Todennäköisenä syynä pidettiin sitä, että ihmisrobotin ollessa mallina, oli palauteoppimisen toteuttaminen helpompaa. Joukkueen, joka koostui pelkistä tavallisista roboteista, kehittämä kielioppi muodostui sattumanvaraisesti. Joukkue, jossa oli mukana oikean ihmisten käyttämän kieliopin osaava robotti, muodosti aina yhteisen kielioppinsa näiden oikeiden sääntöjen mukaiseksi. Ne oppivat käyttämään järjestystä <esine><suhde><esine> ja sanomaan ”Book on the right of the ball” sen sijaan, että sanoisivat syntaktisesti väärän muodon ”Book ball on the right of the”.

Tutkimus osoittaa, että palauteoppimisen hyödyntäminen on tehokas tapa oppia kieltä myös syntaktisen tiedon kannalta. Asetelma oli yksinkertainen ja sisälsi vain vähän muuttujia. Monimutkaisen ympäristön voi olettaa lisäävän kielellisen yhteisymmärryksen syntymiseen tarvittavien toistokertojen määrää, mutta mekanismi säilyy samana, vaikka suhteita ja esineitä lisättäisiin [MaD15]. Yrityksen ja erehdyksen periaate johti tutkimuksessa lopulta, satunnaisia poikkeustapauksia lukuun ottamatta, yhteisen kielipelin löytymiseen, eli kaikki osapuolet oppivat kommunikoimaan toistensa kanssa kokeessa syntyneellä kielellä.

4.3 Hierarkkinen palauteoppiminen

Palauteoppiminen on erinomainen keino löytää todennäköisin oikean tuloksen tuottava ratkaisu yksittäiseen tilanteeseen huomioiden samalla taustamuuttujia. Kielen monitasoisuuden, muuttuvuuden ja olemassa olevien sanojen

määrän takia toimintojen ja tilojen yhdistelmät kasvavat kuitenkin todella suuriksi. Tätä koneoppimiselle ominaista ongelmaa kutsutaan dimensiokirokseksi (engl. curse of dimensionality). Jos esimerkiksi yritetään mallintaa yhteisjakauma kymmenelle peräkkäiselle sanalle luonnollisessa kielessä 100000 sanan sanavarastolla, on mahdollisia parametreja potentiaalisesti 100000-1 = 10-1 [BeY08]. Ongelmaa on yleisesti yritetty ratkaista joko vähentämällä tilojen viemää kapasiteettiä tai käyttämällä edullisia opetustekniikoita, kuten simulaatioiden avulla opetusta [RiO11].

Hierarkkinen palauteoppiminen on uudenlainen ratkaisumalli dimensiokirouksen ongelmaan. Menetelmässä ongelmat jaetaan tarpeen mukaan pienempiin osaongelmiin, jolloin jokaisen osaongelman sisältämästä tietomäärästä tulee käsiteltävissä olevan kokoinen. Tehtävät jaetaan useille agenteille, jotka mallinnetaan yhden ison markovin prosessin sijaan pienistä osatehtävistä huolehtivilla semi-Markov-prosesseilla (engl. semi-markov decision process). Agentit voidaan määritellä itsenäisiksi vain omasta alatehtävästään huolehtiviksi toimijoiksi tai niiden päätökset voivat osittain perustua tietoihin kokonaistehtävän tilanteesta [DeN15].

Koska hierarkkinen palauteoppiminen jakaa tehtävät osaongelmiin, voidaan sen avulla ottaa huomioon keskustelun konteksti ja muuttuva ympäristö. Semi-markovin prosessit huolehtivat alatehtävistä, jotka ovat toimintamalleja tietyssä kontekstissa. Hypoteettinen ihmistä jäljittelevä puhuva ostostentekijärobotti voisi esimerkiksi saapuessaan kaupan kassalle siirtyä suorittamaan tilanteesta vastaavaa alatehtävää. Tehtävästä vastaava semi-markovin prosessia käyttävä agentti on oppinut aineistosta ja harjoittelun avulla kaupan kassalla asioinnin kielipelin. Toteutuksen mukaan agentti voi toimia täysin erillisessä tilassa tai sillä voi olla jotakin tietoa koko kauppareissun kulusta. Sen voisi olla hyvä esimerkiksi kassalla asioinnin yhteydessä voida ilmoittaa ostosten keräämisen aikana ilmi tulleista tuotteiden puutteista.

Hierarkkisen palauteoppimisjärjestelmän agenttien tehtävät voidaan jakaa myös pienempiin osiin ja huolehtimaan eri tilanteen sijaan kielen eri tasoihin liittyvistä tehtävistä. Cambridgen yliopiston julkaisussa “Hierarchical reinforcement learning for situated natural language generation” raportoitiin tutkimuksesta, jossa esiteltiin hierarkkiseen palauteoppimiseen perustuva luonnollista kieltä tuottava järjestelmä [DeN15]. Järjestelmä sai kommunikoinnin lähtötiedot ihmisten välisistä keskusteluista kootusta kieliaineistosta ja oppi ajan myötä löytämään sopivan suhteen tehokkuuden ja yksityiskohtaisuuden välillä ihmiselle ohjeistuksia antaessaan.

Tutkimusta varten toteutetussa harjoituksessa järjestelmän tehtävä oli neuvoa ihmistä mahdollisimman selkeitä ja luonnollisia ilmaisuja käyttäen navigoimaan virtuaalisessa 3D-mallinnetussa ympäristössä. Harjoituksessa käytettiin 14 eri oppivaa agenttia, joiden suorittavat tehtävät jakautuivat sisällön valintaan, ilmaisun semantiikan suunnitteluun ja lopullisen ilmaisun valintaan. Agentit pystyivät välittämään tietoa keskenään toimintojen tuottamien tilamuutosten muodossa. Koska palauteoppimista käyttävä agentti

vaatii paljon kanssakäymistä ennen kuin tulos alkaa olla optimaalinen, tutkimuksessa käytettiin simuloitua ympäristöä oikeiden ihmiskäyttäjien sijaan. Simulaation tekemisessä käytettiin hyväksi edellisten samassa ympäristössä toteutettujen harjoitusten kieliaineistoa. Järjestelyn avulla agentti pääsi kokeilemaan montaa eri toimintastrategiaa eri ympäristöissä. Tämä mahdollisti parhaan strategian löytämisen yrityksen ja erehdyksen periaatteella.

Hierarkkisessa mallissa agenttien alatehtävien tuottamat palkinnot kasautuvat yhteen lopulliseksi pitkän aikavälin palkinnoksi. Testissä käytettiin niin kutsuttua kolmeulotteista palkintofunktiota. Funktion ulottuvuudet eli kolme palkintoja tuottava kategoriaa olivat: käyttäjien tyytyväisyys, ihmisen kaltaisten ilmaisujen käyttö sekä ilmaisujen johdonmukaisuuden ja vaihtelun tasapaino. Funktiossa käyttäjän tyytyväisyyttä mitattiin GTS-asteikolla, jolla arvioitiin käyttäjän toimimista ohjeiden mukaan. GTS-asteikossa on binäärisen arvostelun (onnistui tai ei onnistunut) sijaan neljä onnistumisen astetta. Ilmaisujen luonnollisuutta arvioitiin todennäköisyyteen perustuvalla menetelmällä vertaamalla niitä kieliaineistoon. Ilmaisujen johdonmukaisuuden ja vaihtelun tasapainoa arvioiva funktio oli ilmaisunvalinta-agentti, joka etsi kieliaineistosta tilastollisia menetelmiä käyttäen ilmaisujen vaihtamisen säännönmukaisuuksia ja tapoja. Lopulta nämä kolme funktiota yhdistettiin laskemalla yhteen niiden tuottamat palkinnot, aina kun tilanteeseen soveltui useampi kuin yksi ulottuvuuksista.

Järjestelmää testattiin lopuksi vielä ihmiskäyttäjillä. Ihmiset arvioivat pääosin järjestelmän tuottamat ilmaisut luonteviksi ja ymmärrettäviksi. Ihmiskäyttäjät arvioivat tietoa jakaneiden hierarkkisten agenttien tuottamat ohjeet paremmiksi kaikissa kategorioissa, vaikka erot eivät olleetkaan tilastollisesti merkittäviä. Tutkijat arvioivat tutkimuksen tulosten perusteella, että hierarkkinen palauteoppimisen malli sopii hyvin luonnollisen kielen tuottamiseen ja sopii varsinkin eri kokoisiin ja vaihtelevan määrän muuttujia sisältäviin tilanteisiin, koska jakaa kompleksiset ongelmat alatehtäviin. Lisäksi hierarkkinen malli hyödyntää aikaisempia malleja kokonaisvaltaisempaa ja useampia eri näkökulmia huomioon ottavaa palkintofunktiota.

5 Palauteoppiminen ja ympäröivä maailma

Palauteoppimista voi hyödyntää kielen tuottamisessa monella tavalla ja monissa eri vaiheissa. Palauteoppimisen periaatetta voi soveltaa kielen eri aspektien opetteluun. Esimerkiksi kielen syntaksin oppimiseen vertailu sekä yrityksen ja erehdyksen periaate toimivat erinomaisesti. Toisaalta Markovin mallin mukaiset tila-toiminta-palkinto -kuvaukset soveltuvat hyvin esimerkiksi tilannesidonnaisen dialogin opetteluun.

Seuraavaksi tarkastellaan kahta näkökulmaa, joilla palauteoppimisen hyödyllisyyttä voi perustella. Ensimmäinen on yhteys ihmisen oppimismalliin ja toinen palauteoppimiselle ominainen kyky reagoida muuttuvaan toimintaym-

päristöön.

5.1 Yhteys ihmisen oppimismalliin

Monitieteellisessä julkaisussa Brain and Language julkaistussa tutkimusartikkelissa "Reinforcement Learning in Young Adults With Developmental Language Impairment."[LeJ12] vertailtiin erityisvaikeuksista kärsivien nuorten ja tavallisten nuorten kielen omaksumista ja ihmisaivojen käyttämää palauteoppimista. Tutkimuksessa todettiin huomattava yhteys kielen omaksumisen ja palauteoppimisen mekanismien välillä. Huonommin kieltä oppivien nuorten huomattiin myös hyödyntävän huonommin palauteoppimista muiden asioiden omaksumisessa.

Vaikka oppimistavat ovat eroavia, liittyy niihin yhtäläisyyksiä ja samoja aivojen alueita, erityisesti oppimisesta tunnetusti vastaavat tyvitumakkeet. Molempiin liittyvät assosiaatioiden kehittäminen ja jatkuva ärsykkeistä oppiminen, kuten esimerkiksi shakinpelaajan tapaan oppia todennäköisesti parhaaseen tulokseen johtava siirto tai 2-vuotiaan ensimmäisten sanojen oppimiseen. Lisäksi kumpikin oppimistapa on alitajuinen toiminto ja oppiminen tapahtuu niin sanotusti automaattisesti. Tähän perustuu ihmisen tuntema intuitiivinen päätöksenteko ja lapsen kyky oppia kieltä nopeasti ja vaivattomasti ilman muodollista opetusta. Lisäksi molemmissa oppimistavoissa ovat tärkeässä osassa menneiden tapahtumien tuomat kokemukset ja valintojen odotettu vaikutus tulevaisuuteen. Mielihyvän tuntemuksia välittävä hormoni, dopamiini, esimerkiksi säätelee kokemuksien opettamia toimintatapoja. Sen tuotanto lisääntyy, kun odotettavissa on positiivinen palkinto ja vähenee, kun toiminta tuntuu johtavan negatiiviseen lopputulokseen [LeJ12].

Nämä palauteoppimisen ja ihmisen kielen omaksumisen mallien yhtäläisyydet ovat hyvä peruste olettaa, että palauteoppimisen mallintaminen on tehokas tapa kehittää luonnollista kieltä käyttävä tekoäly. Luonnollista kieltä voi käyttää ilmaisuvoimaisimmin, kun sen on oppinut äidinkielenään, minkä vuoksi pienen lapsen kielenoppimista mahdollisimman paljon jäljittelevä mekanismi soveltuu tehtävään erityisen luontevasti.

5.2 Palauteoppiminen muuttuvassa ympäristössä

Ympäristö, jossa luonnollista kieltä tuotetaan, on usein jatkuvassa muutoksessa. On otettava huomioon, että myös ilmaisun valinta ja muotoilu saattavat muuttaa tätä ympäristöä. Esimerkiksi keskustelutilanteessa valitun ilmaisun hyvyyteen vaikuttaa se, mitä vaikutuksia sillä on keskustelun etenemiseen [RiO11]. Tästä syystä aina ei voi määrittää, mikä tietyssä tilanteessa on paras menettelytapa, ennen kuin on saatu kokemusta siitä, miten se muokkaa ympäristöä ja tulevaisuuden toimintamahdollisuuksia.

Kuten luvussa 3.2 todettiin, palauteoppimisen yksi oleellinen piirre on sen ominaisuus tavoitella pitkän aikavälin palkintoa. Kun kokemusta on kertynyt

tarpeeksi, osaa palauteoppimista käyttävä järjestelmä valita tilanteeseen sopivan toiminnon lopullista palkintojen lopputulosta ajatellen, eikä vain ahneesti välitön palkinto mielessään toimien. Esimerkiksi keskustelussa positiivisen vastaanoton tuottava myönnytys saattaa heikentää asetelmia myöhemmässä keskustelussa. Shakin pelaajan on samoin siirrollaan joskus uhrattava nappula, muuttaakseen shakkilaudan asetelmaa edukseen [RiO11]. Palauteoppiminen kykenee siis ottamaan huomioon kielellistä kommunikaatiota sisältävän tilanteen kokonaisuutena ja oppimaan sitä kautta asiayhteyksien merkityksen ilmaisun valinnassa.

Palauteoppimisen toimintaan kuuluu myös aliluvun 3.2 lopussa mainittu tapa vaihdella toimintansa perustetta todennäköisimmän vaihtoehdon valinnan ja uuden vaihtoehdon kokeilemisen välillä. Tämä ominaisuus auttaa löytämään uusia parempia tapoja toimia. Monipuolinen tieto eri toimintatavoista on tärkeää tilanteessa, jossa ympäristö muuttuu tuottaen uusia tilanteita ja lisää yhteyksiä asioiden välille. Satunnaisen kokeilemisen todennäköisyyteen vaikuttavan muuttujan arvo kannattaa pitää suurempana alkuvaiheessa, jotta opettelussa voidaan ottaa huomioon mahdollisimman useita eri toimintavaihtoehtoja. Kun agentti on oppinut tarpeeksi, voidaan muuttujan arvoa pienentää opitun tiedon tehokkaaksi hyödyntämiseksi. Arvoa ei kannata kuitenkaan laskea nollaan, jotta ympäristön muutoksiin reagoiminen toteutuu.

Joskus ympäristö muuttuu perustaltaan niin paljon, että aikaisemmin hyväksi todettu toimintamalli ei enää sovellukaan tilanteeseen. Järjestelmä voi esimerkiksi olla oppinut kommunikoimaan kielellisesti tiettyjen ihmisten kanssa ja yhtäkkiä keskustelukumppanit vaihtuvat. Vaikka kieli määritelmällisesti olisi sama (esimerkiksi suomen kieli), kielellinen ympäristö muuttuu, samoilla sanoilla voi olla hieman eri merkitys ja eteen saattaa tulla ilmaisuja, jotka eivät ole ennalta tuttuja. Eri keskustelijoilla on käytössään erilaiset kielipelit. Ihmiset kohtaavat elämässä tällaisia tilanteita vieraillessaan uudessa kulttuurillisessa ympäristössä, kuten vaikka eri murrealueella tai kaupunkilainen maalaisympäristössä ja toisinpäin.

Tällaisen muutoksen tapahtuessa (Markovin prosessia käyttäen mallinnettu) palauteoppiva agentti saattaa alkaa esimerkiksi saada toiminnasta a tietyssä tilanteessa s palautteeksi uuden tilan s', joka poikkeaa samassa tilanteessa saman toiminnon aikaisemmin yleensä tuottamasta tilasta s*. Tapauksen toistuessa agentti päivittää tilatietojoukkoon T tilanvaihdosfunktionsa P(s'|s,a) ja P(s*|s,a) lisäten uuden ja vähentäen vanhan tapahtuman odotettua todennäköisyyttä. Palkintofunktio R ilmoittaa tilan tuottaman palkinnon ja vaikuttaa siihen, minkä verran uudet todennäköisyydet aiheuttavat muutoksia agentin käytäntöön (engl. policy).

Edellä kuvatulla tavalla palauteoppimista hyödyntävä agentti reagoi ympäristön muutoksiin ja muuttaa tarvittaessa toimintamalliaan saavuttaakseen kommunikatiivisen tavoitteensa kielen avulla. Tällainen valmius on tärkeä olla olemassa, jotta kyetään huomioimaan myös tavalliset kielessä tapahtuvat

muutokset. Ihmiset kehittävät arkista kieltä jatkuvasti, ja eri kulttuurien ja ideoiden vaikutus muuttaa vanhoja sanoja silloin tällöin korvaten niitä uusilla. Oppimisen täytyy siis olla jatkuvaa, eikä luonnollisen kielen osaamisessa voi tulla valmiiksi.

6 Yhteenveto

Luonnollinen kieli on ihmisten käyttämää kieltä, jonka tuottaminen on vaikeaa sen monimutkaisen ja muuttuvien sääntöjen vuoksi. Näitä kaikkia sääntöjä on lähes mahdotonta kirjoittaa loogisina lausekkeina ohjelman tietoon, mutta palauteoppimiseen perustuva järjestelmä voi ne yrityksen ja erehdyksen periaatteen avulla omaksua. Palauteoppiminen on koneoppimisen tekniikka, joka hyödyntää samanlaista oppimisen mallia kuin kieltä opetteleva ihminen. Luonnollisen kielen tuottaminen on ihmiselle ominainen taito, minkä vuoksi oppimismallin jäljittely on perusteltua. Oppimismallin valintaa tukee myös sen ominaisuus ottaa huomioon pitkän aikavälin palkinto ja mahdollisuus päivittää tietojaan jatkuvasti ympäristöstä saatavan palautteen perusteella.

Palauteoppimista voi hyödyntää kielen eri tasoihin liittyvän tiedon omaksumiseen. Markovin päätöksentekoprosessilla voidaan mallintaa keskustelu ja valita mihin tarkoitukseen sopivia ilmaisuja kannattaa käyttää missäkin tilanteessa. Prosessin avulla voidaan opetella siis Wittgensteinin korostama ilmaisun käyttötapa kielessä, eli sen merkitys.

Hierarkkinen palauteoppiminen kykenee ottamaan huomioon asiayhteyden ja purkamaan kielen tuottamiseen liittyvät tehtävät osatehtäviksi pienentäen yksittäisen agentin käsittelemien tilojen ja toimintojen määrää. Se mahdollistaa monimutkaistenkin keskustelun tuottamiseen liittyvien ongelmien käsittelyn tehokkaasti. Alaluvussa 4.3 esitellyn hierarkkisen palauteoppimisen mallin käyttämä palkintofunktio osasi ottaa huomioon useamman kielellisen ulottuvuuden ja osaltaan kehittää järjestelmän tuottamaa kieltä. Alaluvussa 4.2 esitelty robottien ympäristön kuvaamistehtävä taas osoittaa, miten palauteoppimisen avulla voidaan opettaa kielioppisääntöjä eli kielen syntaksia.

Esimerkkilause ja lähdeviite [4]. [5] [3] [7] [1] [2] [3] [6] [8] [9] [10] [11] [12] [13]

Lähteet

[1] Arsith, Mirela: *Ludwig Wittgenstein and Language Games (A Literary Application).* Acta Universitatis Danubius. Communicatio, 5(2), 2011.

[2] Bellman, Richard: *A Markovian decision process.* tekninen raportti, DTIC Document, 1957.

[3] Bengio, Y. ja Senecal, J. S.: *Adaptive importance sampling to accelerate training of a neural probabilistic language model.* IEEE Transactions on Neural Networks, 19(4):713–722, huhtikuu 2008. https://search.proquest.com/docview/70475557?accountid=11365.

[4] Dethlefs, N. ja Cuayáhuitl, H.: *Hierarchical reinforcement learning for situated natural language generation.* Natural Language Engineering, 21(3):391–435, 2015. https://www.scopus.com/inward/record.uri?eid=2-s2.0-84930152603&doi=10.1017%2fS1351324913000375&partnerID=40&md5=038aed145428f0426dbecb4e5496b2d0, cited By 5.

[5] Khan, Shahzad: *Ethem Alpaydin. Introduction to Machine Learning.* Natural Language Engineering, 14(1):133–137, tammikuu 2008. https://search.proquest.com/docview/216641061?accountid=11365, Copyright - 2007 Cambridge University Press; Last updated - 2015-05-30.

[6] Lee, Joanna C ja Tomblin, J Bruce: *Reinforcement learning in young adults with developmental language impairment.* Brain and language, 123(3):154–163, 2012.

[7] Levin, E., Pieraccini, R. ja Eckert, W.: *Using Markov decision process for learning dialogue strategies.* Teoksessa *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, nide 1, sivut 201–204, 1998. https://www.scopus.com/inward/record.uri?eid=2-s2.0-0031624616&doi=10.1109%2fICASSP.1998.674402&partnerID=40&md5=408ed492e2482618b3899ec91ccd8799, cited By 59.

[8] Littman, Michael L: *Reinforcement learning improves behaviour from evaluative feedback.* Nature, 521(7553):445–451, 2015.

[9] Maravall, Darío, Mingo, Jack Mario ja De Lope, Javier: *Alignment in vision-based syntactic language games for teams of robots using stochastic regular grammars and reinforcement learning: The fully autonomous case and the human supervised case.* Robotics and Autonomous Systems, 63:180–186, 2015.

[10] Mingo, Jack Mario ja Aler, Ricardo: *A competence-performance based model to develop a syntactic language for artificial agents.* Information Sciences, 373:79–94, 2016.

[11] Rieser, Verena ja Lemon, Oliver: *Reinforcement learning for adaptive dialogue systems: a data-driven methodology for dialogue management and natural language generation.* Springer Science & Business Media, 2011.

[12] Schatzmann, Jost, Weilhammer, Karl, Stuttle, Matt ja Young, Steve: *A survey of statistical user simulation techniques for reinforcement-learning of dialogue management strategies.* The knowledge engineering review, 21(2):97–126, 2006.

[13] Sutton, Richard S ja Barto, Andrew G: *Reinforcement learning: An introduction*, nide 1. MIT press Cambridge, 1998.