

Lappeenrannan-Lahden teknillinen yliopisto LUT

School of Engineering Science

Tietotekniikan koulutusohjelma

Kandidaatintyö

**Sami Arho**

**NEUROVERKKOJEN HYÖDYNTÄMINEN E-URHEILUSSA PELAAJAN  
MEKAANISTEN TAITOJEN VALMENNUKSESSA**

Työn tarkastaja(t): Ass. Prof. Ari Happonen

Työn ohjaaja(t): Ass. Prof. Ari Happonen

# TIIVISTELMÄ

Lappeenrannan-Lahden teknillinen yliopisto LUT

School of Engineering Science

Tietotekniikan koulutusohjelma

Sami Arho

**Neuroverkkojen hyödyntäminen e-urheilussa pelaajan mekaanisten taitojen valmennuksessa**

Kandidaatintyö

2020

28 sivua, 6 kuvaa

Työn tarkastajat: Ass. Prof. Ari Happonen

Hakusanat: e-urheilu, tekoäly, koneoppiminen, valmennus

Keywords: e-sports, artificial intelligence, machine learning, coaching

Tässä työssä tutkitaan mahdollisuuksia kehittää tekoälyyn pohjautuva ohjelmisto, jonka avulla voitaisiin joko valmentaa e-urheilun ammattilaisia, tai kilpapelailun harrastajia. Työ keskittyy yrityksen vaatimusten mukaisesti Defense of the Ancients 2 (DotA 2) -pelin vaatimiin mekaanisiin taitoihin, kuten reaktioaikaan tai hiiren sijaintiin. Tutkimustyön lisäksi kehitettiin sovellus, joka analysoi keskiverto pelaajan toimintoja ja liikkeitä ammattilaispelaajan vastaaviin parametreihin. Sovelluksen avulla pystytään antamaan käyttäjälle palautetta siitä, miten hänen toimintojansa tulisi

parantaa ammattilaispelaajaan nähden. Työssä todettiin, että kyseisen valmennusohjelman tuottaminen olisi mahdollista, ja että kyseiselle ohjelmalle olisi käyttöä.

## **ABSTRACT**

Lappeenranta-Lahti University of Technology LUT  
School of Engineering Science  
Degree Program in Computer Science

Sami Arho

### **Utilization of neural networks in e-sports coaching for player's mechanical skills**

Bachelor's Thesis

28 pages, 6 figures

Examiners: Ass. Prof. Ari Happonen

Keywords: e-sports, artificial intelligence, machine learning, coaching

This paper explores the possibilities of developing a software based on artificial intelligence, that could be used to either train professional e-sports players or players enjoying competitive gaming. The paper focuses on the specifications of mechanical skills required by the computer game Defense of the Ancients 2 (DotA 2), such as reaction time or mouse position as required by the company. In addition to the research, an application was developed that analyzes the actions and movements of the average player to the corresponding parameters of a professional player. The application is able to give the user feedback on how his actions could be improved in relation to a professional player. The paper concluded that it would be possible to produce such a software, and that there would be a demand for it.

## **ALKUSANAT**

Työ on tehty Lappeenrannan-Lahden teknillisessä yliopistossa LUT:ssa. Kovan kamppailun ja varsinkin viivästelyiden saattamana. Kiittäisin ensisijaisesti työni tarkastajaa, ja siinä valtavasti auttanutta Ari Haposta. Tämän lisäksi kiitän Kuisma Närheä kandidaatintyön yhteistyöstä.

# SISÄLLYSLUETTELO

1. JOHDANTO .....	8
1.1 Työn taustaa .....	8
1.2 E-urheilun historiaa ja taustaa .....	9
1.3 Defense of the Ancients 2 ja mekaanisen valmennuksen osalta oleelliset termit ja toiminnot .....	9
1.4 Tavoitteet ja rajaukset .....	11
1.5 Työn rakenne .....	12
2. AIKAISEMPI TUTKIMUS .....	13
2.1 OpenAI Five .....	13
2.2 Muu tutkimus .....	13
3. TUTKIMUSMENETELMÄT .....	15
3.1 Tutkimus- ja kehitystyö .....	15
3.2 Peliteoria .....	15
4. KEHITYSTYÖ .....	17
4.1 Työkalut .....	17
4.2 Parametrit .....	17
<b>4.2.1 Parametrien valinta</b> .....	18
<b>4.2.2 Parametrien hyödyntäminen</b> .....	20
4.3 Datan generointi .....	25
5. TULOKSET .....	26
6. POHDINTA JA TULEVAISUUS .....	27
7. YHTEENVETO .....	28
LÄHTEET .....	29

## **SYMBOLI- JA LYHENNELUETTELO**

DotA 2	Defense of the Ancients 2
CS:GO	Counter-Strike Global Offensive
FPS	First Person Shooter
HoN	Heroes of Newerth
LoL	League of Legends
TCG	Trading Card Game

# 1. JOHDANTO

Siinä missä yleisestikin ottaen yhteiskunta on digitalisoitunut ja perinteiset toimijat siirtyvät puhtaista fyysisistä tuotteista tietopalveluilla kuorutettujen ratkaisuiden suuntaan (Kortelainen et al.,2019), niin samalla tavalla myös perinteinen urheilu, omalla tavallaan, kokee digitalisaation vaikutuksia e-urheilun muodossa. Toisaalta keinoäly tekee läpimurtoansa lähes jokaisella yhteiskunnan saralla, toisaalta sitä käytetään hyödyksi yhteiskunta jätteen minimoimiseksi ja tuotteiden kierrätettävyyden parantamiseksi (Ghoreishi & Happonen, 2019), mutta vastaavasti sitä voidaan hyvin soveltaa e-urheiluun ja data pohjaisesti tehostettuun virtuaaliseen valmentamiseenkin [7].

## 1.1 Työn taustaa

Videopelien pelaaminen on monelle pelkkä ajanviete tai harrastus [8], mutta nykyään kilpailuhenkinen pelaaminen on yleistynyt [9] ja tästä syystä monelle pelaamisesta on tullut myös ura. Vaikka keskittyisit tietyn videopelin oppimiseen ja siinä paranemiseen tai harkitset videopelien pelaamista työksesi, on oppimista varten niukasti tehokkaita ja helppokäyttöisiä vaihtoehtoja. Voit seurata ammattilaisten pelaamista useita tunteja ja pyrkiä soveltamaan heidän pelityyliään itsellesi. Tämä kuitenkin vaatisi pelaajalta saman tasoisia reaktioaikoja ja pelitietämystä kuin ammattilaisella, minkä he ovat saaneet kokemuksella. Voisit siis käyttää lukemattomia määriä ajastasi pelin harjoitteluun, mutta tämä ei ole kovin tehokasta, ja alan yleistymisen takia kilpailu on kovaa. Harjoittelua varten on tästä syystä nykyään myös tarjolla useita valmennusohjelmia, joissa voit ostaa esimerkiksi entisten ammattilaispelaajien tarjoamaa valmennusta. Valmennuksen hinta vaihtelee kuitenkin valmentajan mukaan, ja keskiverto valmentajasta saa yleensä maksaa pari kymmentä euroa tunnilta. Kyseessä on kuitenkin keskiverto valmentaja eikä kohderyhmällä, eli nuorilla, ole rahaa maksaa edes tuollaisia summia.

GameMentor Oy on lähtenyt kehittämään ideaa, jossa hyödynnettäisiin ammattilaispelaajien pelaamia pelejä ja koneoppimista siten, että koneäly oppisi ammattilaisten pelaamista peleistä ja



pystyisi analysoimaan parhaan mahdollisen tavan toimia tietyissä tilanteissa ja täten antamaan palautetta käyttäjälle ilman erillistä valmentajaa.

## **1.2 E-urheilun historiaa ja taustaa**

E-urheilu on kilpailullista videopelien pelaamista, jota on harrastettu siitä lähtien, kun ensimmäiset videopelit tuotettiin 1950-luvulla [1]. Kuitenkin 2000-luvulla e-urheilu on lähtenyt nousuun räjähdysmäisesti, ja nykyään katsojia on miljoonittain (Happonen & Minashkina, 2019) ja rahapalkinnot voivat olla useiden kymmenien miljoonien luokkaa. Alan tuottavuus ja suosio ovat johtaneet siihen, että nykyään on todella kova kilpailu pelaajien ja joukkueiden kesken, ja siksi pelaajilla on myös kysyntää erilaisille työkaluille harjoittelua varten [10].

GameMentor Oy kehittää erilaista lähestymistapaa pelaajien valmentamiseen, luomalla uusia työkaluja, jotka kilpailevat hinnallaan, saatavuudellaan ja tehokkuudellaan nykyisten valmennusmetodien kanssa. Tarkoituksena on luoda ohjelmistoja valmentamaan pelaajia erilaisissa e-urheilu-peleissä koneoppimista hyödyntäen. Kandidaatintyössä pohdittava ohjelmisto on alustavasti jaettu erikseen MechCoach- ja StratCoach nimisiin työkaluihin, joista tässä työssä keskitytään MechCoach osuuteen. MechCoachin tarkoituksena on antaa käyttäjälle palautetta pelatuista peleistä liittyen erilaisiin mekaanisiin tekijöihin, kuten hiiren sijaintiin ja reaktionopeuteen. Ohjelmistoa lähdetään kehittämään DotA 2 -pelin pohjalta, mutta tarkoituksena on lähteä laajentamaan ohjelmistoa myöhemmin siten, että sitä voidaan hyödyntää noin 20 %:a suosituimmista e-urheilupeleistä.

## **1.3 Defense of the Ancients 2 ja mekaanisen valmennuksen osalta oleelliset termit ja toiminnot**

DotA 2 on lajityyppinä videopeleissä tunnettu MOBA:na (Taisteluareenamoninpeli), tähän genreen kuuluu DotA 2:n lisäksi myös LoL (League of Legends) ja HoN (Heroes of Newerth), jonka kehittäminen on sittemmin lakkautettu DotA 2:n suosion noustessa. DotA 2 -pelistä otettu

kuvankaappaus näkyy kuvassa 1. Kuvassa nähdään pelille tuttu käyttöliittymä, jossa kolme Radiant-puolen sankaria ovat tappamassa yhtä Dire-puolen sankaria.



*Kuva 1: Kuvankaappaus DotA 2 -pelistä*

Pelissä on muiden pelaajien lisäksi kätyreitä. Näitä kätyreitä ohjaa tietokone, ja kätyreitä tappamalla pelaaja saa joukkueellensa kätyristä riippuen tietyn määrän resursseja, eli kultaa ja kokempisteitä. Kullalla voi ostaa ottelun aikana kaupasta esineitä, jotka vahvistavat omia sankareita, tai antavat esimerkiksi näkyvyyttä ympäri karttaa. Kokempisteitä keräämällä pelaajat saavat tasoja, jotka vahvistavat automaattisesti sankareita, ja antavat heille taitopisteitä, joita voi käyttää taitojen oppimiseen. Taidoilla tarkoitetaan sankareille ominaisia liikkeitä, jotka vaihtelevat hyvin paljon eri sankareiden välillä. Näistä taidoista käydään tässä työssä Monkey King -sankarin osalta myöhemmin kahta.

Pelaajat jaetaan pelin alussa viiden hengen joukkueisiin, joissa kartan vasemmassa alanurkassa on Radiantin tukikohta, ja oikeassa ylänurkassa on Diren tukikohta. Tämän jälkeen pelaajat valitsevat oman sankarinsa 117 eri vaihtoehdosta. Joukkueiden tehtävänä on tuhota vastustajajoukkueen tukikohta, eli Ancient, ennen kuin he ehtivät tuhota oman joukkueen tukikohdan. Kuvasta 2 näkee Ancientin lisäksi, että pelissä on myös muita rakennuksia kuten sitä puolustavia torneja, jotka pitää tuhota ennen kuin pääsee vihollisen Ancientin kimppuun. Muut pelaajat pyrkivät tietysti

puolustamaan myös omaa Ancientiaan parhaansa mukaan. Pelin aikana ja sen edetessä pyritään keräämään mahdollisimman tehokkaasti ja nopeasti resursseja (kokemuspisteitä ja kultaa). Jos toisella joukkueella on enemmän resursseja kuin toisella pelaajien väliset taistelut suosivat geneerisesti sitä puolta, jolla on enemmän resursseja, ja siksi resurssien keräämisen tehokkuus on yksi pelin tärkeimmistä osa-alueista.



*Kuva 2: Kuvankaappaus pelistä, jossa Dire-puolen tukikohta (Ancient)*

## 1.4 Tavoitteet ja rajaukset

MechCoach-ohjelmistoa lähdetään kehittämään alustavasti siten, että keskitytään koneoppimiseen, jonka tavoitteena on opettaa itseään analysoimalla ammattilaisten pelaamien pelien dataa. Analysoimalla tätä dataa ja seuraamalla ammattilaispelaajien tekemien päätösten lopputuloksia koneälyn tulisi pystyä erottamaan hyvät pelaajan tekemät päätökset huonoista. Ohjelmisto pystyy tämän datan avulla heijastamaan käyttäjän pelaamisen ns. optimipelaamiseen ja antaa siten palautetta siitä, miten käyttäjän pelaaminen erosi ammattipelaajasta ja mitä asioita hänen tulisi parantaa.

Työn tavoitteena on kehittää MechCoach-ohjelmistosta MVP (Minimum Viable Product) siten, että keskitytään toistaiseksi vain yhteen sankariin DotA 2 -pelissä. Suunnitteluvaiheessa huomattiin, että oman jäsentäjän teko ohjelmistoa varten olisi liian työläs ja toistaiseksi tarpeeton, sillä työn päällimmäisenä tarkoituksena on luoda koneäly, joka pystyy opettamaan itseään erilaisella peliin liittyvällä datalla. Teknisesti työtä lähdetään siis edistämään kehittämällä ohjelma, joka tuottaa samankaltaista dataa, jota myös lopullinen ohjelmisto käyttäisi. Tätä teetettyä dataa käytetään sitten ohjelmistoa kehitettäessä. Tämän lisäksi luodaan itse koneäly, jonka tulisi opettaa itseään dataa hyödyntäen siihen pisteeseen, että se pystyy antamaan luotettavaa palautetta hyvistä ja huonoista pelauksista.

## **1.5 Työn rakenne**

Aikaisemmassa tutkimuksessa käydään läpi työhön liittyvää kirjallisuutta ja olemassa olevia työkaluja. Tähän käytetään pääosin Google Scholar:n hakukonetta, mutta myös tarvittaessa Googlea. Aikarajauksena tutkitaan kirjallisuutta ja työkaluja 2000-luvulta eteenpäin. Tutkimusmenetelmät käsittelevät toimintatapaa siitä, millä MVP:ia lähdetään toteuttamaan, ja miten tarvittava tieto kerätään ja analysoidaan. Kehitystyössä käydään läpi työssä käytettävät työkalut ja erilaiset parametrit, joita hyödynnetään tekoälyä kouluttaessa ja käydään läpi mistä parametrit tulevat. Tuloksissa tarkastellaan kehitetyn MVP:n tuottamat tulokset ja niiden oikeellisuus. Pohdinta ja tulevaisuus luku käsittelee lopullisen ohjelmiston kehittämisen tulevaisuutta ja pohdintaa kyseisestä tuotteesta markkinoilla. Yhteenveto kokoaa työssä käydyt asiat ja niistä johdetut lopputulokset.

## **2. AIKAISEMPI TUTKIMUS**

Aikaisempaa tutkimusta valittiin tähän lukuun sen olennaisuuden kannalta. Pyrittiin valitsemaan vähäisestä tutkimuksesta huolimatta sellaisia, jotka antaisivat kuvakulmaa ja esimerkkejä siitä, minkälaista tutkimusta aihetta sivuten on löydettävissä. OpenAI Five on ollut mullistava keksintö e-urheilussa, ja se opettelee samantapaista päätöksentekoa, kuin työssä on tarkoitus tutkia.

### **2.1 OpenAI Five**

OpenAI Five perustuu viiteen eri neuroverkkoon, jotka pelaavat DotA 2 pelissä joukkueena. OpenAI kehitti tekoälyä pelauttamalla sitä toisia botteja vastaan uudestaan ja uudestaan, antaen sille pikkuhiljaa käsityksen siitä minkälaiset päätökset auttavat sen mahdollisuuksia voittaa. Nämä päätökset, joita tekoäly oppi ovat myös hyvin lähellä tämän opinnäytetyön päämäärää, vaikkakin data hankitaan eri tavalla.

### **2.2 Muu tutkimus**

Vaikka e-urheilu on nuori käsite siihen liittyvää tutkimusta kuitenkin jo kehitetty suhteellisen paljon [11]. Varsinkin koneälyyn ja pelaamiseen kohdistunut tutkimus on kasvanut räjähdysmäisesti viime vuosina. Esimerkiksi DotA 2:een sovellettu OpenAI Five, joka päihitti ensimmäisenä tekoälynä DotA 2:n puolustavan mestarijoukkueen OG:n (Berner ym. 2019), on nostanut mielenkiintoa ja kysyntää alan tutkimukselle valtavasti.

Muu tutkimus tekoälyn hyödyntämisestä e-urheilussa ja pelaamisessa on ollut vähäistä, vaikka tekoälyä onkin sovellettu jo useilla muilla alueilla. Kuten aiemmin onkin jo todettu tähän, on tulossa muutos, ja riippumatta siitä mihin tekoälyä tullaan e-urheilussa pääosin käyttämään ovat sen tuottamat edut alan kannalta valtavia. Esimerkiksi OpenAI Five:n osalta tekoälyä voidaan

käyttää kehittämään uusia taktiikoita tai sitä voidaan käyttää joukkueiden kouluttamiseen tietyissä pelin osa-alueissa.

### **3. TUTKIMUSMENETELMÄT**

Kappaleessa käsitellään työssä käytetyt tutkimusmenetelmät, ja niiden soveltuvuudet kyseiselle työlle.

#### **3.1 Tutkimus- ja kehitystyö**

Koska opinnäytetyötä koskevaa identtistä tutkimus- ja kehitystyötä ei ole lähes ollenkaan, turvaudutaan kehitystyöhön liittyvässä tutkimuksessa pääosin kehittäjän kokemukseen ja näkemykseen siitä mitkä kehitysmetodit toimisivat parhaiten. Tämän lisäksi kuitenkin etsitään ja käytetään muuta tutkimustyötä, joka tukee kehityksen ideointia ja toteutusta. Nämä lähestymistavat käydään kuitenkin asiakkaan (GameMentor Oy) kanssa läpi ja hyväksytetään ne siten, että myös asiakas näkee lähestymistavat sopiviksi. Jos näin ei ole, käydään asiakkaan kanssa läpi vaihtoehtoiset lähestymistavat, kunnes päästään ratkaisuun, jolla edetään.

#### **3.2 Peliteoria**

Peliteoria käsittelee strategisissa peleissä käytyä kanssakäymistä eri pelaajien välillä, joka maksimoi pelaajien hyödyn ottaen huomioon muiden pelaajien valinnat. Peliteorian mukaan peli käsittelee kolmea pääkohtaa; pelaajia, strategiaa ja tuloksia (Sirkka 2013). Tutkimusmenetelmänä peliteoria toimii hyvin parametrien valintaan tämän opinnäytetyön käsittelemää tekoälyä varten, sillä se tarkastelee optimaalisen toimintatavan pelaajien välillä, ottaen huomioon toiset pelaajat sekä omien valintojen hyödyt ja haitat. Vaikka MechCoach ohjelmisto tulee käyttämään hyvin suoraviivaisia parametreja, tulee kuitenkin parametrien valinnoissa huomioida eri parametrien tärkeys palautetta antaessa.

Peliteorian lisäksi ohjelmiston kehittämisen kannalta on käyty keskustelua asiakkaan kanssa siitä, miten kehitystyötä lähdetään edistämään. Tämän lisäksi on sovittu, että sovellamme myös

kehittäjän olemassa olevaa kokemusta pelistä ja siitä, millä prioriteeteilla ja mitä parametreja valitaan käytettäväksi.



## 4. KEHITYSTYÖ

Tässä kappaleessa käydään läpi ohjelmiston kehittämistä varten vaadittavia työkaluja, parametreja sekä datan generointia.

### 4.1 Työkalut

Ohjelmistoa lähdetään kehittämään Python-ohjelmointikielellä, johon otetaan avuksi tarvittavat kirjastot koneoppimista varten. Python valittiin siksi, että kyseinen ohjelmointikieli on entuudestaan tuttu, ja kielellä oli kehitystyölle valmiit kirjastot tekoälyn kehitykselle. Ohjelmistolle tärkeimmät kirjastot, joita hyödynnettiin tekoälyä luodessa, olivat numpy, lasagna ja theano. Tämän lisäksi datan varastointia varten käytettiin MySQL-kantaa. Python-kieli valittiin pääosin kehittäjien, mutta myös asiakkaan yhteisymmärryksestä sen yleiskäyttöisyyden ja projektiin soveltuvuuden takia. Alustavan tutkimuksen kautta päädyttiin käyttämään Lasagnaa ja siihen liitoksissa olevia theanoa ja numpyä. Tässä vaiheessa keskityttiin siihen, että kyseisiä kirjastoja varten löytyisi kattavat ohjeistukset niiden käyttöä varten, sillä kehittäjillä on ollut ennen tätä työtä suhteellisen vähän kokemusta tekoälyn kehittämisestä. Tämän lisäksi tutkittiin, että kirjastot soveltuisivat hyvin kehitettävää ohjelmistoa varten.

### 4.2 Parametrit

Riippuen mistä pelistä puhutaan e-urheilussa tarvitaan erilaisia taitoja, jotkut pelit vaativat nopeita reaktioita ja peliälyä ja toiset pelit saattavat vaatia pelkkää strategiaa ja arviointikykyä. E-urheilu ei siinä mielessä eroa perinteisistä urheilulajeista, että etenkin huipputasolla tarvitaan myös hyvää peruskuntoa ja terveyttä (Happonen & Minashkina, 2019), minkä lisäksi jokaisessa pelissä on omat perinteisen urheilun kaltaiset lajikohtaiset erikoisuutensa, joissa tietyt taidot, pelaajan ominaisuudet, ajattelumallit, opitut käytänteet yms. joko tuovat pelaajalle etua tai ovat jopa hänelle

haitaksi. MOBA pelit eroavat esimerkiksi TCG (Trading Card Game) peleistä siten, että ne vaativat hyvin pitkälti kaikkia osa-alueita tasapuolisesti, toki riippuen siitä mitä roolia pelaa joukkueessa. Jotkut pelit ja pelityypit painottuvat tietyille osa-alueille, kuten esimerkiksi TCG Hearthstone, vaatii hyvin pitkälti pelkästään strategiaa ja arviointikykyä. Nämä tietysti vaikuttavat siihen minkälainen data on tärkeää tietyille pelille, sillä variaatioita on paljon.

DotA 2:ssa on yleisesti sovittuna kolme eri vaihetta, jotka ovat kaistavaihe, keskipeli ja loppupeli. Seuraavat määritelmät on otettu karkeasti ja kiteytetysti erilaisten DotA 2 -peliin liittyvien oppaiden tekijän ja selostajan Kevin ”Purge” Godec ylläpitämiltä sivuilta (<https://purgegamers.true.io/purge/phases-of-the-game/>). Kaistavaihe kestää pelin kellossa nollassa viiteentoista minuuttiin. Tässä vaiheessa pelaajat ovat jakautuneet kolmelle eri kaistalle, joissa pelaajat pääosin pysyvät tuona aikana. Tässä vaiheessa pyritään maksimoimaan oman joukkueen hankitut resurssit ja minimoimaan vastustajan saamat resurssit. Keskipeli on pelinvaihe, jossa pelaajat liikkuvat ympäri karttaa ja aloittavat torneille puskemisen ja niiden tuhoamisen. Tässä pelinvaiheessa alkaa näkyä huomattavasti enemmän joukkotappeluita, joissa on molemmista joukkueista useampia pelaajia. Loppupeli pitää sisällään eniten yhteistyötä ja organisaatiota oman joukkueen kanssa. Tässä vaiheessa pyritään puskemaan vihollisen tukikohtaan pääosin pelkästään 5 vastaan 5 tappeluilla, ja tuhoamaan heidän Ancient.

Ylempänä käydyistä vaihteista tässä työssä kuitenkin keskitytään pääosin kaistavaiheeseen, sillä tässä vaiheessa eri pelaukset ja ylipäänsä mekaniikkojen perusosaaminen on huomattavasti helpommin havainnollistettavissa ja analysoitavissa.

#### **4.2.1 Parametrien valinta**

Mekaanista dataa DotA 2 -pelissä ovat esimerkiksi sankarin-, hiiren-, ja kameran sijainnit sekä reaktionopeudet ja kätyreiden elämänpisteet tietyissä kohdissa. Mekaanisen datan kerääminen DotA 2:ssa on huomattavasti hankalampaa verrattuna pelitilastoihin, sillä mekaanista dataa ei toistaiseksi saa tietyssä pelin kohdassa suoraan erinäisten ohjelmistorajapintojen kautta ulos. Esimerkiksi suuren osan pelitilastoista saa OpenDota API:n kautta käytettäväksi, mutta pelaajan hiiren sijaintia tietyssä tilanteessa ei näiltä rajapinnoilta saa toistaiseksi millään. Työn alkuvaiheissa ideoitiin mekaanisen datan osalta jonkinlaista parseria, joka pystyisi pelejä

prosessoimalla jäsentämään ohjelmistolle tarvittavan datan sille tarpeelliseen käsiteltävään muotoon. Koska parseria ei kuitenkaan vielä tässä MVP-projektissa tuoteta, parametrien valintaan käytetään suurimmaksi osaksi kehittäjän pelikokemuksesta tullutta intuitiota ja löydettyjä ammattilaispelaajien haastatteluja. Tämä vaihe kehitystyössä on äärimmäisen tärkeä. Valinnassa pitää olla tarkkana mitä parametrejä valitsee sillä suuri osa saattaa olla triviaalista, eikä siitä ole loppukäyttäjälle hyötyä ja datan keruu aiheuttaa vain turhia lisäkustannuksia. Lisäksi parametrien valinnassa pitää ottaa huomioon kohderyhmä, ja kehitettävää ohjelmistoa pyritään kohdistamaan ensisijaisesti keskiverto pelaajille. Tämä on tärkeää siksi, ettei analyyseissä välttämättä kannata keskittyä niin paljoa parametreihin kuten esimerkiksi reaktioaikaan, joka on huomattavasti tärkeämpi erottaja korkeammalla tasolla, mutta ei vaikuta paljoa keskiverto peleissä. Tässä työssä ei esitellä kaikkia valittuja parametreja, sillä DotA:n kompleksisuuden takia parametreja tulee olemaan useita kymmeniä. Käsitellyt parametrit tulee ottaa huomioon sekä pelaajan, että vastustajan puolelta. Eri pelaajien liikkeet ja toiminnot vaikuttavat moninpeleissä äärimmäisen paljon siihen, miten itse tulisi tiettyssä tilanteessa käyttäytyä.

Eniten DotA 2:ta pelatessa käytetään hiirtä. Pelaajat liikuttavat sankariaan hiirellä, siirtämällä hiiren paikkaan, johon he haluavat sankarin liikkuvan ja klikkaamalla oikealla näppäimellä. Jos hiiren kohdistaa samoin tavoin vihollisen päälle sankari pyrkii hyökkäämään tähän. Tämän lisäksi hiirtä käytetään (yleensä mutta ei aina) kameran siirtämiseen liikuttamalla hiiri pelin nurkkaan siirtäen kameraa siihen suuntaan. Nämä ovat kuitenkin vain yleisimmät tarkoitukset hiirelle, ja perustasolla tärkeimmät. Näppäinpainalluksia käytetään yleensä taitojen ja esineiden käyttämiseen, joiden kohdistaminen tapahtuu myös hiirellä. Useita muitakin käyttötarkoituksia näppäimille löytyy kuten yksiköiden valitseminen ja pysähtyminen, mutta näppäimiä voi myös sitoa esimerkiksi tiettyihin tärkeisiin paikkoihin karttaa kuten riimujen kohdalle. Hiiren sijainti ja näppäinpainallukset ovat pohjimmasta dataa, jota käytetään. Tämä data analysoidaan, ja sen avulla pyritään selvittämään pelaajan suoriutuminen ja varsinkin epäonnistuminen peleissä.

Keskiverto peleissä geneerisesti tärkeimpinä tekijöinä voi nostaa esiin mm. kätyrien tappamisen, animaation perumisen, taitojen ja esineiden käyttämisen sekä sankarien sijainnin. Kätyrien tappaminen tarkoittaa tässä tilanteessa sitä, että pelaaja saa viimeisen iskun joko vastustajan kätyriin, josta pelaaja ansaitsee kokemuspisteiden lisäksi myös viimeisen iskun tuottaman kullan, tai omaan kätyriin. Omaan kätyriin kohdistunutta hyökkäystä kutsutaan kieltämiseksi, tässä

tilanteessa vastustaja ei tietenkään saa viimeisen iskun tuottamaa kultaa, vaan kieltävä pelaaja saa nykyisen päivityksen mukaan 20% normaalista kätyrin antamasta kullasta ja lähistöllä oleva vastustaja saa myös vain 40% normaaleista kokemuspisteistä. Kuten aiemmin todettu nämä resurssit ovat hyvin tärkeitä pelin etenemisen kannalta, ja pelejä seurattaessa näitä kahta resurssia seurataan joukkueiden välillä pääteltäessä kummalla joukkueella on etulyönti asema.

#### **4.2.2 Parametrien hyödyntäminen**

Ylemmässä kappaleessa käytyjä esimerkkejä eri parametreista voidaan käyttää analysoimaan suurinta osaa pelin tapahtumista. Kaistavaiheessa kuitenkin korostuvat kätyrien tappaminen, taitojen käyttäminen, sekä hyökkäyksen- ja animaation peruminen.

#### **Kätyrien tappaminen**

Kätyrien tappaminen on tärkeä taito jokaiselle pelaajalle, mutta varsinkin niille, jotka pyrkivät keräämään kultaa. Näitä pelaajia kutsutaan kantajiksi, ja heidän tarkoituksenaan on kerätä mahdollisimman paljon kultaa ja kokemusta mahdollisimman nopeasti, sillä heidän sankarinsa ovat vahvimpia loppupelissä ja vaikuttavat eniten voittamiseen tässä vaiheessa. Myös mitä nopeammin nämä pelaajat saavat resursseja, ja varsinkin jos he saavat nopeammin resursseja kuin vastapuolen kantajat, he ovat huomattavasti vahvempia myös pelin aikaisemmissa vaiheissa jolla on suuri merkitys.



**Kuva 3:** *Monkey King saamassa tapon kätyristä*

Vaikka kätyrien tappaminen vaikuttaa hyvin yksinkertaiselta, vaikeutuu prosessi huomattavasti, kun vastaan laitetaan toinen pelaaja. Tämä johtuu työssä aiemmin sivutusta kieltämisestä, joka mahdollistaa pelaajat kilpailemaan molempien puolien kätyreistä. Tästä syntyy pelaajien välille hämäyspeli, jossa pyritään huijaamaan vastustajaa lyömään kätyreihin liian aikaisin, liian myöhään tai pakottamalla vastapuolen pelaaja lyömään omaa sankariaan silloin kun olisi mahdollisuus saada tappo kätyristä. Kätyrien tappamisessa tärkeimpinä asioina mekaanisesti nousevat kätyrien ja sankarien sen hetkiset elämäpisteet, sankarien sijainnit sekä hyökkäykset ja taitojen käyttö. Optimaalisin aika lyödä kätyreitä on yleensä juuri ennen kuin ne olisivat muulloin kuolemassa, sillä tällöin kuoleva kätyri on tehnyt eniten vahinkoa omaan kätyriaaltoon. Tämä on tärkeää siksi, että kätyrit tappelisivat keskenään sinulle suotuisimmassa paikassa, eli mahdollisimman lähellä omaa torniasi kuitenkin niin, ettei torni hyökkää vihollisen kätyreihin. Tässä paikassa vihollisen on hyvin vaikea yrittää tappaa sinua, sillä sinulla on todella lyhyt matka oman tornisi alle turvaan. Tämä tilanne on kuitenkin päinvastainen, kun pyrit kieltämään omia kätyreitäsi viholliselta, ja sinun tulisi aloittaa niihin hyökkääminen heti kuin mahdollista. Omiin kätyreihin voi aloittaa hyökkäämisen, kun kätyrin terveys on tippunut 50 %:iin tai alle. Oman

sankarin sijainnin tulisi kätyrien tappamisessa olla lähes aina mahdollisimman lähellä tapettavaa kätyriä, jotta tapon saaminen helpottuu. Tämä kuitenkin riippuu normaali tilanteesta vahvasti eri seikoista, kuten siitä missä vihollissankarit sijaitsevat, missä kohtaa kätyriaalto sijaitsee tilanteesta ja paljonko elämäpisteitä ja resursseja väylällä olevilla sankareilla on. Kuitenkin yksilöidyssä tilanteessa, jossa ei ole muita sankareita, jokaisessa tilanteessa tulisi olla valmiiksi lähellä kätyreitä ja kääntyneenä kätyreitä kohti. Kätyrin tappaminen vaikeutuu, jos pelaaja joutuu käyttämään ylimääräistä aikaa kääntymiseen tai liikkumiseen.

### **Taitojen käyttäminen**

Sankarien eri taitoja käytetään yleensä hyvin rajallisesti, mutta tämäkin toki riippuu taidosta. Monkey King sankarilla on käytössään kätyrien tappamiseen liittyen kuudesta taidostaan tärkeimpinä kaksi, jotka ovat Boundless Strike ja Primal Spring. Boundless Strike kasvattaa Monkey King:in sauvaa, Monkey King:n lyöden sen maahan eteensä tainnuttamalla ja vahingoittamalla vihollisia.



**Kuva 4:** *Monkey King:in Boundless Strike*

Boundless Strike on näistä taidoista selkeästi tärkeämpi kätyrien tappamisessa, sillä Primal Spring on huomattavasti hitaampi, tekee vähemmän vahinkoa ja vaatii Monkey King:in olevan puun



päällä ennen sen käyttöä. Lisäksi Boundless Strike tainnuttaa viholliset, joka on hyvin tärkeä kätyreistä taistellessa vihollispelaajaa vastaan.



**Kuva 5:** *Monkey King:in Primal Spring*

Kuten aiemmin mainittiin, näitä taitoja tulee käyttää rajallisesti kätyrien tappamiseen, mutta jos kätyrien viereen ei jostain syystä ehdi tai pääse, on yleensä hyvä idea käyttää näitä taitoja siihen ettei menetä kätyreistä tappoa. Tämän lisäksi taitoja voi käyttää kuitenkin myös vastustajan hännäämiseen tainnuttamalla ja vahingoittamalla heitä, kuitenkin tämä tulisi tehdä lähes aina pelkästään silloin, kun pelaaja saa myös varmistettua ainakin yhden kätyritapon tai vaihtoehtoisesti evättyä yhden kätyritapon viholliselta.

### **Hyökkäyksen- ja animaation peruminen**

DotA 2:ssa sankareilla on tietynlainen hyökkäysanimaatio, joka vaihtelee sankarien välillä. Hyökkäys koostuu kahdesta eri komponentista; hyökkäysanimaatiosta ja taakse viennistä. Jos sankari hyökkää paikoillaan tekemättä mitään, sankari aloittaa hyökkäyksen, jonka jälkeen hyökkäyksen taakse vienti vie tietyn verran aikaa ennen kuin seuraava hyökkäys voidaan aloittaa.

Tätä havainnollistaa Kuva 3, jossa Death Prophet nimisen sankarin hyökkäys on jo lähtenyt mutta takaisin vieni kuluttaa kuvassa turhaa aikaa.



**Kuva 6:** *Hyökkäyksen taakse viennin havainnollistaminen*

Taakse viennin perumisessa periaate on se, että kun hyökkäys on joko kaukohyökkäyksessä lähtenyt tai lähihyökkäyksessä osunut, pyritään käyttämään taakse viennin hukkaan kuluttamaa aikaa hyödyksi liikkumalla sen aikana. Tämä mahdollistaa esimerkiksi vihollissankaria jahdatessa ylimääräisiä hyökkäyksiä, joita ei olisi saanut pelkästään paikoillaan pysymällä. Nämä hyökkäykset ovat tärkeitä varsinkin alkupelissä, kun sankareilla ei ole vielä niin paljoa hyökkäysnopeutta, joka vähentää hyökkäysanimaation ja sen taakse viennin kestoa. Taakse viennin peruminen liikkumalla on yleensä kriittinen tekijä keskiverto pelaajilla, ja sen käyttö vaikuttaa suuresti pelin etenemiseen ja siten sen hyödyntämiseen ja optimoimiseen tulisi käyttää aikaa. Hyökkäyksen perumista voidaan taas käyttää hämäämään vastustajaa esimerkiksi kaistavaiheessa tilanteessa, jossa molemmat ovat iskemässä samaan kätyriin, ja kätyri on kahden hyökkäyksen päässä kuolemasta. Hämäämällä vastustajaa perumalla oman hyökkäyksen vastustaja saattaa lyödä kätyriä itse. Vastustaja luulee tilanteessa saavan viimeisen lyönnin, mutta



koska oma hyökkäys peruttiin, ehditään iskemään viimeinen lyönti itse vastustajan lyönnin jälkeen.

Hyökkäysten ja sen taakse viennin lisäksi sankareiden taitoja voidaan perua. Taitojen perumisessa sankari näyttää siltä, että se on käyttämässä jotain taitoa, mutta viime hetkellä peruu sen, joka estää käytetyn taidon menemistä jäähyllä säästäen myös muita taidon käyttämiä resursseja kuten mana. Taitoja perumalla pyritään yleensä huijaamaan vastapelaajaa eri tavoin. Vastapelaaja voidaan saada käyttämään omia taitojaan, joka laittaa ne jäähyllä ja käyttää pelaajan resursseja, hänen saadessa yleensä mitätön hyöty taidon käytöstä. Vaihtoehtoisesti pyritään saamaan pelaaja väistämään turhaan iskua, jota ei koskaan tule. Tämän taidon avulla on helpompi ennustaa mihin vastustaja liikkuu, tai manipuloida minne pelaaja liikkuu.

### **4.3 Datan generointi**

Tekoälyn koulutusta varten generoitiin data-alkioita aiemmin läpikäydyistä parametreistä yhteensä 10 000 kappaletta. Nämä arvot generoidaan parametrien puitteissa pelille realistisilla arvoilla. Näitä alkioita olivat:

- Hiiren sijainti näytöllä
- Sankarin sijainti
- Kameran sijainti
- Sankarin tekemä vahinko
- Kätyrin elämäpisteet
- Pelaajan elämäpisteet
- Reaktionopeus

Tämän lisäksi generoitiin vielä 1000 kappaletta ylimääräisiä alkioita, joilla validoitiin tekoälyn onnistuminen. Käytetyt luvut ovat samat, kuin katsotun Lasagna-tutoriaalin, jotka tuottavat käytetyllä Python-skriptillä uskottavan yli 90 % tarkkuuden, joka on sovittu myös asiakkaan kanssa riittäväksi.

## 5. TULOKSET

Koneäly saatiin koulutettua käyttämällä generoituja data-alkioita siten, että se antoi eri tilanteissa oikean vaihtoehdon ensin 96.2 %:in todennäköisyydellä ja toisella kerralla 95.8 %:in todennäköisyydellä. Arvoja verrattiin hiiren-, sankarin- ja kameran sijainnissa siihen, missä kohtaa oli kuolemassa oleva kätyri. Sankarin tekemää vahinkoa verrattiin kätyrin elämäpisteisiin siten, olisiko vahingolla saatu viimeinen isku kyseiseen kätyriin. Reaktionopeuksia verrattiin siten, että koneäly valitsi aina pienemmän reaktionopeuden optimaaliseksi. Koneälyä testattiin vielä useamman kerran tämän jälkeen, ja kaikki arvo olivat asiakkaan kanssa sovitun tarkkuuden yläpuolella eli yli 90 %. Eri tilanteista saatujen tulosten, ja niissä onnistumisen mukaan, MechCoach-ohjelmiston luominen vaikuttaa teoriassa mahdolliselta.

Laitteistona toimi oma kotikoneeni, jossa käytettiin kouluttamista varten grafiikkakorttia (Nvidia GeForce GTX 1070). Grafiikkakorttia käytettiin suorittimen sijaan siksi, että Theano -kirjaston avulla käytetty Lasagne pystyy hyödyntämään sitä verkostoa kouluttaessa, ja Lasagnen dokumentaation mukaan se on 10 – 50 kertaa nopeampaa. Grafiikkakorttia hyödyntämällä kouluttamiseen näillä aineistoilla ja resursseilla kului noin 3 tuntia.

## 6. POHDINTA JA TULEVAISUUS

Alustavan tekoälyn ja työn teorian onnistuessa MechCoach-sovellus vaikuttaisi olevan teoriassa mahdollista toteuttaa. Teoriassa siksi, että lopullinen sovellus vaatisi parserin, jonka kehitys vaikuttaa tämän työn perusteella hankalalta ja kyseisen parserin tekeminen voi jäädä esteeksi lopullista sovellusta kehittäessä. Kuitenkin tämä työ käsitteli vain työhön vaativaa alustavaa teoriaa yhden sankarin kohdalla. Raa'an mallin luominen yhdelle sankarille ei kuitenkaan vaikuta kovinkaan isolta työmäärältä käytännössä, mutta pelidatan saamisessa saattaa olla ongelmia. Kuitenkin nykyäänkin löytyy jo valmiiksi DotA 2 -pelistä luotuja verkkosivuja ja työkaluja, joilla saa tietynlaista dataa ulos. Data kohdistuu kuitenkin pääosin pelin strategiseen puoleen ja pelaajien tekemiin valintoihin, mutta mekaanista dataa kuten sankarien sijaintiakin on mahdollista saada ulos. Eri mahdollisuuksia mekaanisen datan keräämiselle tulee kuitenkin tutkia tulevaisuudessa, jos dataa ei ole mahdollista saada käyttöön suoraan. Tulevaisuudessa tulee myös ottaa huomioon useiden eri pelaajien ja sankareiden tuomia muutoksia ja kanssakäymisiä, jotka monimutkaistavat työtä paljon.

Tällä hetkellä työlle löytyy kuitenkin selvä markkinarako, ja tulevaisuudessa kysyntää tulee olemaan varmasti vielä enemmän käytännölliselle ja edulliselle vaihtoehdolle.

## 7. YHTEENVETO

Työssä lähdettiin kehittämään MVP:tä koneälyyn perustuvasta valmennusohjelmasta DotA 2 - peliin nimeltä MechCoach. Tähän liittyen käytiin läpi pelin sääntöjä ja strategiaa, sekä eri parametreja, joita tulisi ottaa huomioon MechCoach ohjelmistoa kehittäessä. Työhön liittyen ei löytynyt paljoa vastaavanlaista tutkimusta, ja tämänkaltaisen valmennusohjelman luominen myös muita pelejä varten vaatii enemmän tutkimustyötä. Kehitystyönä luotiin pelkistynyt versio tulevasta koneälystä, jolle syötettiin käsitellyistä parametreista realistisia arvoja koneälyn opettamista varten. Tämän jälkeen testattiin koneälyn oppiminen syöttämällä uusia arvoja ja vertailemalla niitä toisiinsa siten, että koneäly onnistuisi tunnistamaan aiemmin määrittämällä (>90%) todennäköisyydellä pelaajalle suotuisan vaihtoehdon. Tämän pääosin teoriapohjaisen työn perusteella voidaan siis sanoa, että kyseinen työ olisi mahdollista toteuttaa.

# LÄHTEET

- [1] What Was the First Video Game, Who Invented It and Why [Verkkodokumentti]. (Viitattu 28.04.2020)  
<https://plarium.com/en/blog/the-first-video-game/>
- [2] Kortelainen, H., Happonen, A., Hanski, J. (2019), "From asset provider to knowledge company - transformation in the digital era", In Lecture Notes in Mechanical Engineering, ISSN: 2195-4356, pp. 333-341, doi: 10.1007/978-3-319-95711-1\_33
- [3] Ghoreishi, M., Happonen, A. (2019), Key Enablers for Deploying Artificial Intelligence for Circular Economy Embracing Sustainable Product Design: Three Case Studies, In the 13th International Engineering Research Conference (Eureca 2019), 27.11.2019, Subang Jaya, Malaysia, p. 17
- [4] Happonen, A., Minashkina, D., 2019, Professionalism in Esport: Benefits in Skills and Health & Possible Downsides, ISBN 978-952-335-374-9, ISSN-L 2243-3384, ISSN 2243-3384, p. 36, doi: 10.13140/RG.2.2.28958.59208
- [5] Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemysław "Psyho" Dębiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, Rafal Józefowicz, Scott Gray, Catherine Olsson, Jakub Pachocki, Michael Petrov, Henrique Pondé de Oliveira Pinto, Jonathan Raiman, Tim Salimans, Jeremy Schlatter, Jonas Schneider, Szymon Sidor, Ilya Sutskever, Jie Tang, Filip Wolski, Susan Zhang (2019) "Dota 2 with Large Scale Deep Reinforcement Learning"
- [6] Sirkka, T. 2013 "Peliteoriaa." Tampereen Yliopisto.
- [7] Harald Geissler, Helmut-Schmidt University, Hamburg, Germany  
Melanie Hasenbein, University of Applied Sciences Munich and Fresenius  
University of Applied Sciences, Germany  
Stella Kanatouri, Helmut-Schmidt University, Hamburg, Germany  
Robert Wegener, University of Applied Sciences Northwestern  
Switzerland/Helmut-Schmidt University, Hamburg, Germany (2014) "E-Coaching: Conceptual and Empirical Findings of a Virtual Coaching Programme"
- [8] Uukoski, O. 2011 "Playing video games: A waste of time... or not?" University of Helsinki
- [9] Chikish, Y., Carreras, M., García J., 2019, "eSports: A New Era for the Sports Industry and a New Impulse for the Research in Sports (and) Economics"
- [10] Hamari, J., Sjöblom, M., "What is eSports and why do people watch it?"

[11] Steinkuehler, C., “Esports Research: Critical, Empirical, and Historical Studies of Competitive Videogame Play” 2019

[12] OpenAI Five, <https://openai.com/projects/five/>

[13] GameMentor Oy, <http://gamentor.gg/>