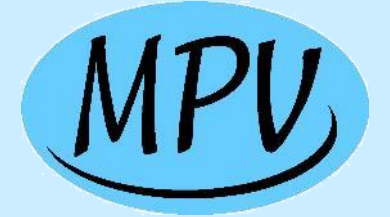


Tekoälyä testauksessa ja hyvän softan teossa



Matti Vuori www.mattivuori.net matti.vuori@mattivuori.net @Matti_Vuori



Sisällysluettelo

<u>Tässä setissä</u>	3
<u>Tekoälyn sovellusstrategiat</u>	4
<u>Tekoäly toiminnan abstraktiotasoilla</u>	6
<u>Käyttökohteita yleisesti</u>	7
<u>Älyn tarvitsee seurata toimintaa</u>	8
<u>Mahdollisuuksia: Yleinen tiedonhaketuki</u>	9
<u>Mahdollisuuksia: Testien suunnittelu</u>	10
<u>Mahdollisuuksia: Kohteen analysointi</u>	12
<u>Mahdollisuuksia: Ongelmien tunnistus</u>	13
<u>Mahdollisuuksia: Bugien korjaus</u>	14
<u>Mahdollisuuksia: Testien suorittaminen</u>	15
<u>Mahdollisuuksia: Monitorointi</u>	16
<u>Mahdollisuuksia: Testauksenhallinta</u>	17

Tässä setissä

- Näinä aikoina kiinnostaa se, mihin kaikkeen tekoälyä voisi käyttää softan testaamisessa ja hyvän softan tekemisessä.
- Tässä setissä on siihen liittyviä ideoita.

Tekoälyn sovellusstrategiat 1/2

- Roolistrategia:
 - Ei etsitä ylivertaista älyä, vaan ylivertaista apulaista.
 - Piilotetun ymmärryksen sijaan välinettä, joka paljastaa asioita ihmiselle.
 - Tekoäly voi esim. esittää tarinan ”Big Datasta”
- Yksinkertaistus/kognitiivinen strategia:
 - Tekoäly apuna pitämään asiat ihmiselle yksinkertaisena.
 - Esim. koodin optimointi käsin vielä 90-luvulla tuottaa kamalaa koodia – kun kääntäjä optimoi, voidaan keskittyä ymmärrettävyyteen.
 - Jos tekoäly optimoi testisetit ajoa varten, voidaan ne rakentaa logiikkapohjaisesti jäsentäen, jolloin kokonaisuus on selkeä.

Tekoälyn sovellusstrategiat 2/2

- Orgaaninen toiminnan kehittämisen strategia:
 - EI: Mihin voisi pultata tekoälyn kiinni, vaan...
 - Missä työtä voisi tehdä:
 - Paremmilla tiedoilla?
 - Muun maailman tietoja ja kokemuksia hyödyntäen?
 - Ongelmia automaattisesti tunnistaa?
 - Hyviä käytäntöjä tukien?
 - Rutiineja ja tylsää työtä automatisoiden?
 - Tunnistamalla erilaisia kaavoja käyttäytymisessä, datassa?
 - Automaattisesti lopputulosta simuloimalla, testaamalla ja arvioimalla?
 - Missä olisi tarve oppivalle ja ahkeralle työkaverilla, mutta ihmiseen ei ole varaa?

Tekoäly toiminnan abstraktiotasoilla

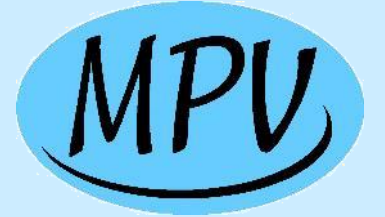
Ihmisen työn taso	Tekoälyn rooli ihmisen kanssa
Ammatti (occupation)	Ammatin identiteettiin kuuluva väline(metsurin moottorisaha)
Työ (job)	Väline, valvoja, rutiineista huolehtija, infran hallinta, sparraaja
Tehtävä (task)	Apulainen, apuväline, työpari, turvatekniikka
Tehtävän vaihe	Apulainen, apuväline, työpari, turvatekniikka, tarkastaja, testaaja
Toiminto	Työkalu, väline, turvalaite, automaattisesti toimiva, tarkastaja, testaaja

Prosessin taso	Tekoälyn rooli prosessissa
Liiketoimintaprosessi	Valvoja, orkestroija / koordinaattori, metriikat, laskenta, suunnittelun tuki
Käytäntö	Oppiva apulainen, tietotuki, päätöksenteon tuki, analysoija
Työnkulku	Autonominen toimija, varmistaja, konfiguraattori, implementoija, suorittaja
Menetelmä	Automaatti, älykäs oppiva suorittaja, ihmisen apulainen, räätälöinti kontekstiin
Tekniikka	Tehostaja, automaattinen suorittaja

Käyttökohteita yleisesti

- Käyttöön, toimintaan liittyvän tiedon **analysointi**. – lokit...
- Monitorointi ja **kaavojen** tunnistaminen – kun tällaisessa tilanteessa esiintyy tällaista, kohta softa kaatuu...
- Artefaktojen analysointi – suunnitelmat, disainit, arkkitehtuurit...
- **Virheiden ja anomalioiden diagnoosi** – mistä asiat johtuvat.
- Asioiden **optimointi** – testisetit...
- Asioiden **generointi** – testit, testidata...
- Ihmisen **tietotuki**. Asiantuntijajärjestelmät.
- Tekemisen **ennakointi ja valmistelu** välineissä.
- Älykäs **tekeminen** – ohjelmistorobototin tekemä testaus

Älyn tarvitsee seurata toimintaa



- Älyn pitää oppiakseen voida seurata kaikkea ja yhdistää sitä muuallakin tapahtuvaa...
- Siksi se suosii kaiken tallentamista:
 - Kaiken mahdollisen automaattinen lokitus.
 - Asioiden dokumentointi.
 - Tiimin puheen kuuntelu.
 - Ilmeiden ja eleiden seuraaminen kameralla.
- => **Tietoturvallisuus, tallennetun datan väärinkäyttö (esim. työntekijöiden valvonnassa...)**

Mahdollisuuksia: Yleinen tiedonhaketuki

- Avaintietojen poiminta dokumenteista, keskusteluketjuista, lokeista.
- Älykäs tiedonhaku useista erilaisista lähteistä.
 - Bugikannat, keskustelupalstat.
 - Relevanttien tietojen tunnistus (termit, konteksti) ja hyvyyden arviointi (kirjoittajan maine, peukutus, ”tietueen” luokittelu, kirjoitustyyli...).
- Automaattinen kontekstin ja tarpeen tunnistus ja haku taustalla tai keskusteleva tietotukijärjestelmä.

Mahdollisuuksia: Testien suunnittelu 1/2

- Parempaa tietoa testauksen suunnitteluun.
 - Hyvää dataa historiasta, testauksen kohteesta.
- Selväkielisten skenaarioiden ”extraktointi” lokeista.
- Mallipohjaisen testauksen testimallien generointi käsityöstä puoliautomaattiseksi.
 - Mitä toteutuksesta löytyy.
 - Miten ohjelmaa voisi käyttää.
 - Kaiken mahdollisen kokeilu.
 - Käytöstä opitun hyödyntäminen testien generoinnissa ja testiajoissa.
- Testiskriptien mutatointi.
 - Rungosta datan variointia, polkujen vaihtelua.

Mahdollisuuksia: Testien suunnittelu 2/2

- Älykäs datan fuzzaus.
- Heuristinen TDD-testigeneraattori.
 - Hyviä testejä.
- Testimallin luominen toteutetusta järjestelmästä – nykyisiä älykkäämmin.
- Kuormitustestausprofiilien rakentaminen.
- Regressiotestisettien rakentaminen muutosten vaikutusanalyysin pohjalta.
- Simuloidun testiympäristön luominen käsin koskematta Big Datan pohjalta.
 - Realismia, yllätyksiä.
 - Muut ohjelmat, järjestelmät / laitteet / IoT.
 - Tapahtumat.

Mahdollisuuksia: Kohteen analysointi

- Arkkitehtuurin arviointi historiadatan perusteella.
- Vaatimusmäärittelyjen ja speksien analysointi.
 - Ongelmat, ristiriidat, puutteet.
- Ohjelman muutosten vaikutusanalyysi.
- Ulkoisten vaatimusten muutosten vaikutusanalyysi.
- Buildien analysointi.
 - Voiko julkaista.
 - Automaattinen selkokielineen release notes.

Mahdollisuuksia: Ongelmien tunnistus

- Bugien ja potentiaalisten ongelmien tunnistus koodista.
 - Opitun bugidatan perusteella.
 - Perinteisiä koodintarkastusohjelmia (Lint) paremmin.
- Ongelmien analysointi
- Juurisyyanalyysi.
- Mallipohjaisen testauksen testilokien analysointi.
- Käytettävyysongelmien tunnistus lokeista ja kuvaus selkokielellä.
 - Toimintojen peruminen tai sahaus toimintojen välillä.
- Sosiaalisen median tekstianalyysi.
 - Mistä puhutaan tuotteen yhteydessä. Tunnelma. Valitukset.

Mahdollisuuksia: Bugien korjaus

- Viallisen koodinpätkän korvaus ehjällä pätkällä toisesta ohjelmasta.
- Tietotuki – älykäs, suodattava ja valikoiva tekstihaku ongelmasta.

Mahdollisuuksia: Testien suorittaminen

- Älykkäät testausrobotit UI-testaukseen.
 - Suuntaa-antava käytettävyytestaus.
 - Bugien etsimisen taktikat.
 - Käyttäjien simulointi.
- Älykkäämpi systeemin monitorointi testauksen aikana.
 - Anomaliat suorituskykymittareissa, toimintojen nopeudessa jne...
- Testaajan vireystilan valvonta ja apu sen hallinnassa.
- Työtilan ilmapiirin älykäs tuki (valot, musiikki...).
- Viestintäapu.
- Bugiraportointiapua osaamattomille, vieraskielisille...

Mahdollisuuksia: Monitorointi

- Testikohteen älykäs monitorointi.
 - Milloin mittareissa (CPU, muistin käyttö) näkyy aiemmin jotain indikoivaksi todettuja kaavoja.
 - Taustalla kaikenlaisessa testauksessa tutkivasta testauksesta pitkäaikaistestaukseen.

Mahdollisuuksia: Testauksenhallinta

- Testisettien / testiajojen optimointi.
 - Priorisointi, järjestys, nopeus.
- Testauksenhallinnan erilaisten tehtävien automatisointi.
- Tehtävien valmistelu.
 - Äly oppii, että tietyssä vaiheessa, tiettynä päivänä tehdään tietynlaisia asioita -> laitetaan työkalut valmiiksi, vihjataan asiasta.
- Datan generointi
- Kaiken logitus.
 - Logien analysointi, automaattiset hyvät bugiraportit.