Tekoäly- ja ohjelmistorobotiikkakokeilun loppuraportti - syksy 2020

Sovellus allekirjoitusten tunnistamiseen PDF-asiakirjoista

Helsinki

Tukea digitalisaatiokokeiluihin kaupungin työntekijöille

Sovellus allekirjoitusten tunnistamiseen PDF-asiakirjoista

Niko Latvakoski (Kymp / Maka / Aska)
Aleksander Alafuzoff (Siili Solutions Oy)
Susa Eräranta (Kymp / Maka / Aska)
Katariina Hirvonen (Kymp / Maka / Aska)
Juuso Ala-Outinen (Kymp / Maka / Aska)
Saska Lohi (Kymp / Hatu / Kepa)

Tietosuoja-asioiden osalta myös:
Jussi Vaanola (Kymp / Hatu)
Tetti Kunnas (Kymp / Hatu)

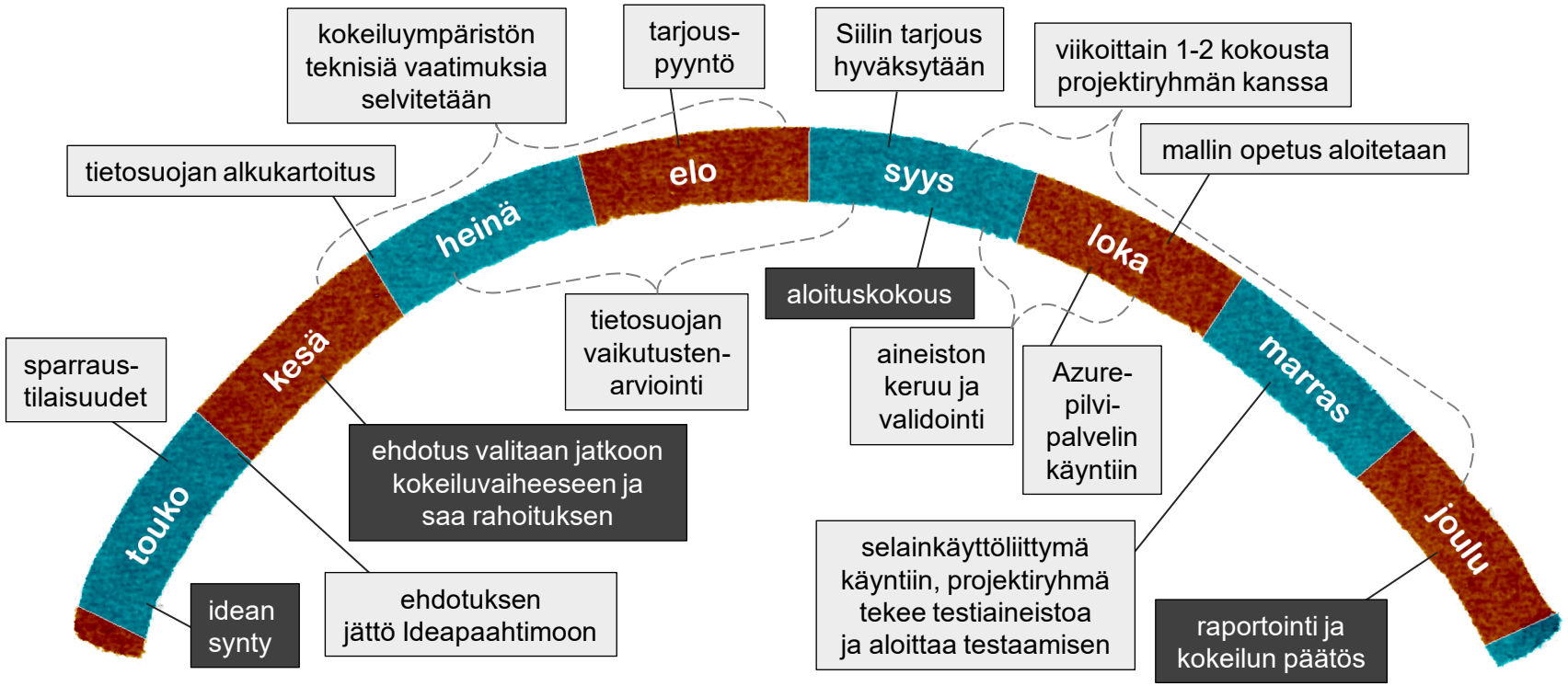
Helsinki

1. Kokeilun onnistuminen

- Tavoitteena oli kehittää työkalu, joka automaattisesti tunnistaa PDF-asiakirjoista allekirjoitukset. Lisäksi haluttiin saada selville, kuinka kuvantunnistukseen perustuva tekoäly käytännössä toimii, ja millaisia seikkoja tekoälyn opettamisessa tulee ottaa huomioon. Lisäksi tavoitteena oli oppia ohjelmistokehityksestä ja projektinhallinnasta.

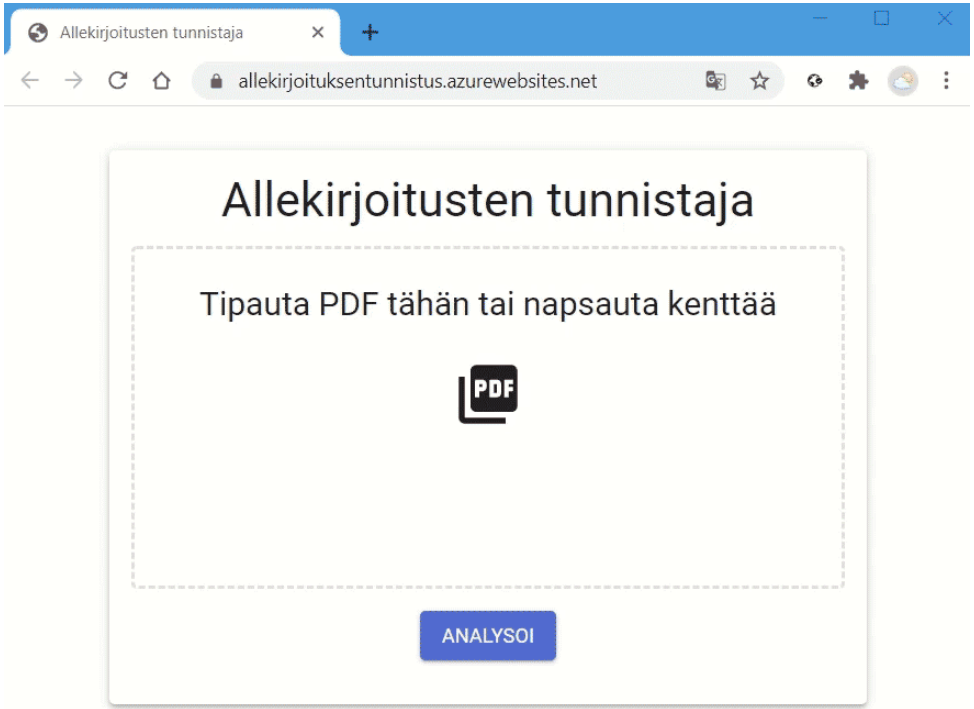
- Tavoitteet saavutettiin erinomaisesti. Tuloksena oli toivotun kaltainen sovellus, jota on mahdollista kehittää ja laajentaa varsinaisen kokeiluhankkeen päätyttyäkin. Kokeilussa opittiin paljon syvien neuroverkkojen toiminnasta, opettamisesta ja testaamisesta.

2. Kokeilun eteneminen



Helsinki

3. Kokeilun tuotokset



Helsinki

4. Opit tekoälyn mahdollisuuksista

- Allekirjoitukset on mahdollista tunnistaa kuvantunnistukseen perustuvan tekoälyn avulla.
- Tekoäly voi toimia tehokkaana tukena asiakirjojen käsittelyssä.
- Asiakirjojen käsittelyyn kuluvaa työaikaa on mahdollista lyhentää.
- Vastaavin keinoin voidaan ehkä tunnistaa muitakin henkilötietoja.

Helsinki

5. Opit tekoälyn kehittämisestä

- Opetus- ja validointiaineisto määrää pitkälti lopputuloksen. Aineiston keruuvaiheessa on syytä keskustella laajasti sen edustavuudesta ja tehdä vertailua.

- Kehityksen aikana aineiston valikoinnissa ja käsittelyssä voi tapahtua oivalluksia ja muutoksia, sen mahdollisuus täytyy jättää myös jatkokehitykselle.

- On hyvä perehtyä siihen, mikä on kokeiluun parhaiten soveltuva, (mahdollisesti esiopetettu) tekoälyalgoritmi.

- Testaaminen erilaisilla, haastavilla aineistoilla auttaa tunnistamaan, mitkä ovat tekoälyn rajat. Rajoitteet on tunnettava, jos halutaan siirtyä tuotantokäyttöön.

- Hyvät tietosuojakäytännöt muistettava joka käänteessä. Tekoälyn opetus on tehtävä tietosuojalainsäädännön ehdoilla.

6. Opit kokeilemisesta

- Ongelman rajaaminen pelkkiin allekirjoituksiin oli sopiva tutkimuskysymys ja haaste.
- Toteutuskumppanin asiantuntemusta kannattaa kuunnella. Työaikaa säästyi mm. allekirjoitusten keruussa ja opetus- ja validointiaineiston koonnissa.
- Viikoittaiset tapaamiset hyvä juttu, kaikki osapuolet pysyy kärryillä ja tietävät omat vastuualueensa.
- Tietosuojan vaikutustenarviointi kannattaa aloittaa ennen kokeilun aloittamista, saadaan selville toteutusympäristön, kokeiluaineiston ja tietosuojakäytäntöjen vaatimukset sekä kokeilua että jatkokehitystä ajatellen.
- Pieni stressi uuden äärellä on hyvä, mutta aina kannattaa selvittää kulloisenkin aiheen asiantuntija ja kysyä nopeammin mielipidettä.
- Kokeilusta saatavien hyötyjen arviointiin (tässä tapauksessa mm. ajansäästö) olisi ollut syytä panostaa enemmän ja varhaisemmassa vaiheessa.

Helsinki

7. Opit resursoinnista

- Ajankäyttö yhteensä 200-250 tuntia koko tiimiltä (aloituskokouksesta 18.9. loppuraporttiin). Ajankäyttö ylittyi vain vähän suunnitellusta.

- Budjetissa pysyttiin, Azuresta tarvittiin maksullista lisätehoa mallin opetusvaiheessa ja muussa tutkimuksessa.

- Aikaa säästettiin käyttämällä julkista ja avointa aineistoa, ja valittiin hyvä esiopetettu algoritmi. Tällaista dataa kannattaa käyttää niin paljon kuin mahdollista tietosuojalainsäädännön rajoissa.

- Resursointi on järkevää tehdä siten, että tarvittaessa henkilötyöaikaa voidaan liittää esim. testaukseen ja hyötyjen arviointiin (projektiryhmän sisältä tai muualta)

Helsinki

8. Kokeilun tekninen ympäristö

- Toteutusympäristö Azure-pilvipalvelin
  - Neuroverkkomalli opetettiin fastai-kirjaston avulla
  - Sovellusta ajetaan Azure Web Apps-palvelun päällä
    - Allekirjoituksentunnistus API (Flask-sovellus)
    - Web-käyttöliittymä (React-sovellus)
  - Sovellus on paketoitu Dockerilla, jotta se voidaan tarvittaessa siirtää Azuresta esim. kaupungin omaan palvelinympäristöön

- Tietosuojasyistä kaupungin sisäverkon laitteistoa (koneet, palvelimet) ja pilvipalveluiden käyttöä koskevia linjauksista olisi hyvä keskustella jo kokeilun ideointivaiheessa.

9. Kokeilun data

- Aineistona käytettiin julkisia, kaupungin internetsivuilta ladattuja asiakirjoja, jotka olivat allekirjoituksen tunnistamisen kannalta edustavia (allekirjoitukset oli pyyhitty pois).

- Allekirjoitusaineisto ladattiin Wikimediasta ja sopivat allekirjoitukset validoitiin (viereinen kuva).

- Sopivia neuroverkkomalleja opetettiin Jupyter Notebooks-ohjelmointityökalulla opetusaineiston pohjalta ja suoritusta arvioitiin validointiaineistolla. Sovellusta on testattu monenlaisilla testitiedostoilla.

Helsinki

10. Jatkopäätökset ja -ideat

- Tuloksia pyritään esittelemään toimialan muutostoimistolle, ICT:lle ja Ahjo-kehittäjille. Oma toivomuksemme on, että kokeilusta otettaisiin koppia ja hyödyt saataisiin siirrettyä koko kaupungin tasolle. Jatko-suunnitteluun osallistujista, rahoituksesta yms. ei ole ollut vielä puhetta tässä vaiheessa.

- Mallin opettamista ja muuta jatkokehitystä pyritään jatkamaan kaupungin sisäisessä palvelinympäristössä ja jatkamaan mallin opetusta. Tästä on tehty / tehdään dokumentaatio ja muut valmistelut.

- Jatkokehityksessä edelleen keskeisenä huomioitavana seikkana tietosuoja, mahdollinen tarve käyttäjähallinnalle (ylläpitäjän rooli) sekä lokitus.

Helsinki