TEKOÄLYPILOTIN RAPORTTI

CASE TVO-AI

Tämä dokumentti on osa Tampereen yliopiston GPT-Lab-tutkimusryhmän toteuttamaa Teknologiayritysten modernisointi tekoälyn ja robotiikan avulla (temotero) -hanketta, jossa pilotoidaan tekoälysovellusten kokeiluja ja Proof of Concept (PoC) -pilotteja satakuntalaisten yritysten tarpeisiin.

Tämä raportti käsittelee pilottia Teollisuuden Voima Oyj:lle. Sisällöstä on poistettu/anonymisoitu se osa sisältöä, joka olisi luottamuksellista ja salassapitosopimuksen (NDA) alaista.

Euroopan unionin osarahoittama

Tampereen yliopisto

GPT-Lab

SATAKUNTALIITTO
Regional Council of Satakunta

Sisällysluettelo

Raportista 3

Modernisointipalvelusta 3

Pilotointiin liittyvät valmistelutoimet 5

Pilotille tarvittavat aineistot 5

Tekoälyagentin tehtäväkuvauksen kehitys 7

Tekoälysovellus 8

Testaus ja pilotointi 12

Hanketiimin tekemien testien tuloksia 12

Yrityksen edustajien suorittaman pilotoinnin tuloksia 14

Analysointia ja jatkokehitysideoita 15

Tekoälyagentin toiminnan analysointia 15

Tekoälyagentin toiminnan jatkokehittäminen 18

Pilotointiin ja yhteistyöhön liittyvät huomiot 21

Yhteystiedot 23

LIITTEET 24

LIITE A (tekoälyagentin tehtäväkuvaus) 24

Raportista

Tässä raportissa kuvataan Tampereen yliopiston (TAU, jatkossa hanketiimi) (osa)toteuttamassa *Teknologiayritysten modernisointi tekoälyn ja robotiikan avulla (temotero)* -hankkeessa tuotettu pilotti Teollisuuden Voima Oyj:lle (TVO, jatkossa yritys).

Pilotissa kokeiltiin ja testattiin tekoälysovelluksen kyvykkyyttä yrityksen data-aineiston käsittelyssä. Hanketiimi toteutti pilottia varten tekoälysovelluksen prototyypin, jossa tekoälyagentin tehtävänä on käydä sille annettua aineistoa läpi ja etsiä aineistosta käyttäjän haussa antamille nimiketiedoille mahdollista vastaavuutta aineistossa.

Pilotissa käytetyn tekniikan ja siihen liittyvän kehitystyön lisäksi raportissa kuvataan pilotoinnin aikana tehtyä yrityksen ja yliopiston tutkijaryhmän välistä yhteistyötä, jossa testataan temotero-hankkeessa suunniteltua modernisointipalvelua ja sen prosessia.

Modernisointipalvelusta

Teknologiayritysten modernisointi **tekoälyn ja robotiikan avulla** -hankkeessa suunnitellaan ja testataan satakuntalaisten yritysten tarpeisiin vastaava modernisointipalvelu, jossa erityisesti pk-yritykset voivat teettää eri alojen asiantuntijoiden ja/tai asiantuntija-/opiskelijatiimien tekemiä kokeiluja ja Proof of Concept (PoC) -pilotteja.

Hankkeen toteuttajina toimivat Satakunnan ammattikorkeakoulu (SAMK) ja Tampereen yliopiston GPT-Lab-tutkimusryhmä Porissa. Modernisointipalvelun prosessia kehitetään hankkeen toiminta-ajan aikana. Prosessin pääpiirteet on esitetty seuraavassa kuvassa (kuva 1).



Kuva 1, Modernisointipalvelun prosessin pääpiirteet

Yhteistyön ja kehityksen teemana ovat robotiikka, automaatio, tekoäly (ja analytiikka). Yhteistyön muodoissa kokeillaan erilaisia yhdistelmiä yrityksen tai organisaation edustajien, yliopiston ja korkeakoulun asiantuntijoiden sekä opiskelija ryhmien muodossa. Yhteistyön toteutuspaikkana voivat toimia esimerkiksi robotiikan tai tekoälyn kehitysympäristöt tai työtä voidaan tehdä yrityksen tiloissa. Prosessin avulla saatu tulos toimii yrityksen tai organisaation jatkotoimien perustana.

Prosessin sisältöön liittyviä täsmennyksiä ja kokemuksia kerrytetään hankkeessa tuotettujen kokeilujen ja pilottien kautta esimerkiksi hankeorganisaatioiden keskinäisillä yhteistapaamisilla.

Kohdeyritys

Tässä pilotoinnissa mukana oleva yritys toimii energiatuotannon alalla ja turvallisuus sekä toimintavarmuus on otettava huomioon kaikessa yrityksen toiminnassa ja prosesseissa. Tämä aiheuttaa myös omat haasteensa toiminnan prosessien kestolle ja kustannustehokkuuden parantamiselle.

Modernisointitoimien kohde (Tarpeen tunnistaminen)

Modernisointikohteen tunnistamiseksi ja pilotointikohteen valintaa varten käytiin keskusteluja yritystä kiinnostavista kehityskohteista. Näissä keskusteluissa yrityksen ja hanketiimin asiantuntijoiden kanssa tunnistettiin viisi (5) mahdollista kehityskohdetta. Hanketiimi vertaili kohteita niiden toteuttamiseen liittyvien työmäärien kannalta, ja yritys puolestaan arvioi kohteita niiden omalle liiketoiminnalleen mahdollistaman potentiaalin näkökulmasta. Jatkokeskusteluissa näkökulmia vertailtiin ja kohteeksi valittiin yhteismielin kohde nro 2 ”teollisuustuotteen tunnistaminen”.

Valitussa kohteessa tekoälyn käytön tavoite olisi auttaa yritystä tunnistamaan omiin rakennuksiinsa liittyvästä laajasta nimiketietoaineistostaan sellaisia kohteita, joita voidaan hankkia suoraan avoimilta markkinoilta ja toimittajilta ilman tarvetta tehdä erikoistilauksia ja niihin liittyviä esiselvityksiä tai erillisten turvatarkastusten teettämistä. Tunnistuksen tuloksena syntyisi uudenlainen aineiston luokittelu (*teollisuustuote*), ja onnistuessaan tekoäly voisi sujuvoittaa yrityksen toimintaprosesseja merkittävästi.

Yhteistyön muoto (Toteutuskokoonpanot)

Yhteistyön muodoksi valikoitui meneillä ollut Temotero-hanke, jossa tarjolla olut maksuton pilotointi tehtäisiin yrityksen edustajien ja hanketiimin asiantuntijoiden yhteistyöllä. Pilotin tyyppinä on Proof of Concept (PoC) -pilotti, jossa todennetaan konseptin toimivuutta käytännössä.

Pilotin toteutuspaikka (Modernisointipalvelun muodot)

Pilottiin sisältyi paljon yhdessä tehtävää suunnittelua ja testausta. Pilotissa käytetty järjestelmä sijaitsisi hanketiimin tiloissa, mutta sitä voitaisiin käyttää verkon yli yrityksen tiloista käsin.

Pilotin tuottamat tulokset (Jatkotoimet)

Pilotin testien tuloksista koottiin analyysikooste sekä ehdotuksia toiminnan jatkokehittämiselle. Mahdollisia jatkotoimia voivat olla esimerkiksi modernisoinnin toteuttaminen pilotin pohjalta tai tilattuna toimittajalta, opinnäytetyön teettäminen tai jatkokokeilu. Tämän tapauksen jatkotoimista yritys päättää erikseen eikä niistä ole kirjauksia tässä raportissa.

Pilotointiin liittyvät valmistelutoimet

Pilotille tarvittavan aineiston osalta tehtiin selvityksiä aineistotarpeista ja niiden saatavuudesta. Tekoälysovelluksen osalta selvitettiin tarvittavat laiteresurssit ja käyttöön soveltuvat mallit.

Varsinaisen tietoaineiston käsittelyä varten sovittiin erillisestä salassapitosopimuksesta hanketiimin ja yrityksen kesken.

Yhteistyön kokonaiskestoksi arvioitiin kaksi kuukautta (touko-kesäkuu). Toteutus kuitenkin venyi teknisten viivästysten ja lomakausien takia elokuun loppuun.

Pilotille tarvittavat aineistot

Tekoälyratkaisut tukeutuvat käytössä olevaan aineistoon ja sen sisältämään dataan ja informaatioon. Pilotissa käytetty aineisto koostuu tekoälyn sisäisestä koulutusaineistosta, yrityskohtaisesta aineistosta sekä hanketiimin pilotin käyttöön hankkimasta tarveaineistosta.

Tekoälyn sisäinen koulutusaineisto

Generatiivisen tekoälyn toiminta perustuu siinä käytetyn kielimallin sisältämään opetusaineistoon. Aineisto sisältää koosteita erilaisista tekstisisällöistä, esimerkiksi kysymys-vastaus-pareina. Aineistonsa myötä tekoäly osaa käsitellä sille annettuja kysymyksiä ja luoda ihmiselle ymmärrettäviä vasteita. Tekoäly ”arvaa” matemaattisen menetelmän avulla tuottamansa tekstin. Aineistonsa myötä tekoäly saa myös substanssinsa, eli ymmärryksen esimerkiksi termeistä ja vakioista. Koska eri kielimallit koostuvat määriltään ja sisällöiltään erilaista opetusaineistoista, vaikuttaa se myös kunkin tekoälyn kykyyn käsitellä sille esitettyjä asioita. Kielimallien sisältämästä opetusaineistosta ei yleensä ole saatavilla yksityiskohtaista tietoa, joten tekoälyn omaan substanssiin ja ymmärrykseen tulee suhtautua varauksella.

Tekoälyn käytössä on myös huomioitava myös se seikka, että monet palveluntarjoajat voivat käyttää tekoälyn kanssa käytettyjä keskusteluja ja aineistoja sekä niiden sisältämiä tietoja uuden koulutusaineistonsa keräämiseen. Tässä pilotissa tahattoman koulutusaineiston keräämisen riskejä on minimoitu käyttämällä paikallisesti ylläpidettyä tekoälyratkaisua sekä palveluita, joissa palveluntarjoaja lisenssisopimuksen mukaan pidättäytyy koulutusaineiston keräämiseltä.

Yrityksen pilottiin tuottama aineisto

Pilottia varten yritys luovutti hankkeen käyttöön nimiketietoaineistoaan. Aineisto sisältää yrityksen tuotantolaitoksissa käytettyjä laitteita ja tarvikkeita. Pilotissa on keskitetty pääaineistosta esierotettuihin osa-aineistoihin eli pientuoteryhmiin ”Tiivisteet”, ”Releet” ja ”Ruuvit”.

Aineistossa on useita tietokenttiä. Pilotissa yrityksen antamasta nimikedatasta tekoälyn tekemiin hakuihin on käytetty seuraavia tietokenttiä:

”A”, *“B”, “C”* ja *”D” (varsinaiset kenttien nimet ja niiden määritykset on julkisversiossa piilotettu)*

Kentät ja niiden kuvaukset on esitetty seuraavassa taulukossa (taulukko 1).

Taulukko 1, Yrityksen nimiketietoaineiston tietokenttiä

KENTTÄ	SELITYKSET		
”A”	<kuvaus>		
”B” ”C”	<kuvaus>	<kuvaus>	<kuvaus>
”D”	<kuvaus>		

Muita aineistossa mukana olevia kenttiä ei ole käytetty. Teknisesti sovelluksessa ei ole mitään esteitä muidenkin kenttien käyttämiseen, mutta pilotin kannalta valitut kentät ovat riittävät.

Hanketiimin pilottiin koostama aineisto

Pilotin alkuperäisenä ajatuksena oli tutkia yrityksen käyttöön antamia nimikeaineistoja suoraan eri toimittajien verkkosivuilta. Hanketiimin tehtävänä olisi ollut luoda tekoäly, joka osaa selata sille annettuja sivustoja. Hanketiimin tekemän esiselvityksen mukaan verkkosivuille kohdistetut haut jättävät jälkiä verkkosivujen tietokantoihin. Kokonaisuutena tiedoista voi syntyä sellaista informaatiota, josta voi kolmansien osapuolten käsissä syntyä uusi turvallisuusriski. Riskien minimoimiseksi aineistoa ei jaettaisi toimittajien verkkosivustoille. Pilotin suunnitelmaa muutettiin niin, että toimittajien aineistot tuotaisiin pilotointiympäristöön ja selaus/haku tehdään siellä.

Pilotointiin tarvittavien aineistojen saamiseksi hanketiimi teki kartoituksen saatavilla olevista tuoteluetteloista ja hinnastoista liittyen edellä määritettyihin pientuoteryhmiin ”Tiivisteet”, ”Releet” ja ”Ruuvit”. Aineistoja etsittiin tunnettujen kotimaisten toimittajayritysten verkkosivuilta. Kartoituksen yhteydessä havaittiin, että aineistojen saatavuudessa on rajoitteita. Osa toimittajista tarjoaa pääsyn aineistoonsa vain käyttäjätunnuksen omaaville tai ei anna pääsyä aineistoonsa ilman yhteystietojen rekisteröintiä. Osalla toimittajista aineisto on saatavilla pienissä osissa.

Kartoituksen reunaehtona (lisäys) materiaalien tuli olla avoimesti ja maksutta saatavilla (ei rekisteröintiä tai käyttölupaa tms. aineiston saamiseksi) ja niiden tulisi olla tekstimuodossa. Kartoituksen tuloksena löytyi useita pilotointiin soveltuvia tuoteluetteloja, joista pilottiin valittiin neljän toimittajayrityksen aineistoja. Toimittajat ja heidän aineistonsa sekä niihin liittyvät kokotiedot on esitetty seuraavassa taulukossa (taulukko 2).

Taulukko 2, Toimittajayritysten tuoteluetteloaineistoja

Toimittaja	Aineisto
Phoenix	Releet ja optokoplerit sekä logiikkamoduulit, 68 sivua / 4 830 riviä
Onninen	Teollisuuden putkistotuotteet (pultit ja mutterit, kierreosat ja putkiliittimet, yms.), 220 sivua / 38 778 riviä
Tiivistekeskus	Tiivisteitä (akselitiivisteet, o-renkaat, yms.), 432 sivua / 119 004 riviä
Würth	Kiinnitystarvikkeet (ruuvit, mutterit, tapit yms.), 134 sivua / 19 764 riviä

Jokaisen mukaan valitun toimittajan aineisto koottiin yhdeksi PDF-tiedostoksi, josta luotiin valmisteluprosessin myötä tekoälyn käyttöön tekstimuotoista dataa. Aineistossa mahdollisesti kuvien muodoissa esitetyt tiedot putoavat tässä vaiheessa prosessia pois.

Pilotoinnin suorituskykyä varten aineiston määrä on tarkoituksella rajattu. Aineistomäärät voivat olla myös suurempia, jos käytössä on enemmän laskentatehoa tai voidaan tyytyä tehtyjen hakujen hitaampiin vasteisiin. Aineistoa on mahdollista esikäsitellä myös poistamalla siitä tarpeettomia sisältöjä tai muuttamalla kuvasisältöjä tekstiksi.

Tekoälyagentin tehtäväkuvauksen kehitys

Tekoälyagentille on luotu tehtävänkuvaus, jossa tekoäly valmistellaan käsittelemään käyttäjän datahakuja. Varsinaiset datahaut ovat yksittäisiä nimiketietorivejä ilman muita käyttäjän tekemiä lisäyksiä. Tekoälyagentti vastaa hakuihin sille annetun tehtäväkuvauksen mukaisesti.

Tekoälyn tehtävänkuvausta (kehote, prompt) kehitettiin erillisillä kehitysiteraatioilla ja niistä syntyneillä versioilla. Kussakin iteraatiossa (versiossa) tehtävänkuvausta testattiin, korjattiin tai tarkennettiin paremman lopputuloksen saamiseksi. Kehitysiteraation työvaiheet on esitetty seuraavassa kuvassa (kuva 2).



Kuva 2, Tekoälyntehtäväkuvauksen kehitysiteraatio

Tehtäväkuvauksen tuottaminen koostuu viidestä työvaiheesta: 1) Tehtävän/kehotteen suunnittelu, jossa määritetään miten tekoälyn tulisi toimia, 2) Tehtävän/kehotteen asettaminen sovellukseen (tai vastaavaan palveluun), 3) Kyselyjen (hakujen) testausta, 4) Hakuvasteiden sisällön analysointia ja tarkistusta, 5) Muutostarpeiden tunnistaminen ja muutosten suunnittelu

Työvaiheista syntyy versiokohtaista dataa. Sovellukselle kehitetty tehtäväkuvaus (viimeisin versio) on esitetty raportin liitteessä (LIITE A).

Kehitysiteraatioiden tuloksena tehtäväkuvauksen määrityksessä on pyritty huomioimaan tietokoneen tapa käsitellä ohjeen sisältöä ja ohjeen koko sekä sen rakenteen selkeys pieniä paikallisia kielimalleja huomioiden. Tekoälyagentin tehtäväkuvaus koostuu seuraavista elementeistä:

1) Tehtävä
2) Päätös
3) Ehdot ja huomautukset
4) Rakenne (kysely ja vaste)
5) Vastauksen kieli ja vaste-esimerkkejä

Kulloinenkin versiotieto kirjataan määrityksen teknisiin muuttujiin, josta se otetaan mukaan käyttöliittymän aloitusnäkymään.

Tekoälysovellus

Pilottia varten tutkimusryhmä tuotti generatiivisen tekoälysovelluksen, jota ajettiin tutkimusryhmän hankkeelle varaamilla laiteresursseilla sekä valitulla kielimallilla ja vektoritietokannalla. Sovellukselle luotiin myös käyttöliittymä.

Palvelinlaitteisto ja käyttöjärjestelmä

Sovellus on tuotettu tutkimusryhmän käytössä olevalle omalle palvelimelle, jossa on tehokas PC sekä muutama GPU-laskentayksikkö. Tässä tapauksessa palvelimen omalla suorittimella, muistilla tai levytilalla ei ole pilotin kannalta suurempaa käytännön merkitystä. Palvelinresurssit on esitetty seuraavassa taulukossa (taulukko 3).

Taulukko 3, Pilotissa käytettyjä palvelinresursseja

Laite	Tekniset tiedot
LLM-palvelin #1	Dell Precision 3680 (tower -työasema) GPU-yksiköt - GPU 0: NVIDIA Tesla P40 (24 GB VRAM) - GPU 1: NVIDIA L4 (24 GB VRAM) Driver & CUDA: NVIDIA Driver 535.216.01, CUDA 12.2 Model Runtime: Ollama runtime (GGUF)
LLM-palvelin #2	Dell Precision 3680 (tower -työasema) GPU-yksiköt - GPU 0: NVIDIA Tesla P40 (24 GB VRAM)
Sovelluspalvelin	Lenovo ThinkStation P350

Linux-pohjaisen käyttöliittymän päällä toimii Ollama-palvelu ja tietokanta. Ollama-palvelu käyttää automaattisesti kaikkia palvelimella saatavilla olevia GPU-resursseja, ja resurssit loppuessa voidaan siirtyä käyttämään palvelimen omaa CPU & RAM-resurssia. Itse pilottisovellus ja kielimallit (eli LLM:t ja upotemallit) suoritetaan fyysisesti eri palvelimilla.

LLM-palvelin #2 on muuten samanlainen kuin ensimmäinen, mutta palvelimessa on vain yksi GPU. Kyseistä resurssia käytetään mm. samankaltaisuuksien etsintään upotemallien avulla (embed

models). Hajautus voi olla hyödyllistä siinä mielessä, että upotemallien laskenta ei vahingossakaan kilpaile laskentaresursseista kielimallin kustannuksella.

Pilotin sovellus on kirjoitettu Python-ohjelmointikielellä, ja sen suoritukseen tarvitaan tuoreehko Python 3 –alusta sekä PIP-paketinhallintasovellus riippuvuuksien asentamiseen. Sovelluskehitys on tehty versiolla 3.12, mutta myös uudemmat/vanhemmat versiot voivat toimia yhtä hyvin. Testauksessa oleva sovellus ajetaan palvelimella Docker Container -kontin sisällä, jolloin sovellukseen liittyvä sisältöä ja suoritusta voidaan hallinnoida turvallisesti. Esimerkiksi sovellus voidaan käynnistää uudelleen ilman, että sillä on vaikutusta palvelimen muuhun toimintaan.

Kielimallit ja vektoritietokannat

Sovelluksen moottorina on kielimalli (LLM) ja siihen yhteensopiva vektoritietokanta RAG-hakua varten.

Sovelluksen vektoritietokannan pohjana on Faiss-kirjasto sekä upotemalli. Kehitystyön aikana palvelimella kokeiltiin useampaa kieli- ja upotemallia. Pilotin aikana kokeiltuja malleja esitellään seuraavassa taulukossa (taulukko 4).

Taulukko 4, Pilotissa käytettyjä kieli- ja upotemalleja

Kielimalli	Koko	Upotemalli
Llama-Poro-2-8B-Instruct	8B/5 Gt	snowflake-arctic-embed2
Llama-Poro-2-70B-Instruct	70B/40 Gt	snowflake-arctic-embed2
Poro-34B-chat	34B/22 Gt	snowflake-arctic-embed2; bge-m3
llama3.3:70b-instruct-q4_K_M	70B/42 Gt	snowflake-arctic-embed2
llama3.3:70b-instruct-q2_K	70B/26 Gt	snowflake-arctic-embed2
llama3.2	3B/2 Gt	bme-m3, llama3.2

Kielimallin nimi sisältää myös tietoja sen koosta. Esimerkiksi ”8B” tarkoittaa kahdeksaa miljardia parametria. Pienintä llama3.2 kielimallia on käytetty lähinnä sovelluskehityksen aikana, koska sen vasteajat olivat erittäin nopeita, eikä vasteiden oikeellisuudella taikka tarkkuudella ollut merkitystä.

Sovelluksen käytön valmisteluun sisältyy lähdeaineiston vektorointia. Vektoroinnissa annettu aineisto (tuoteluettelot) pilkotaan (tokenointi), ja ajetaan upotemallin läpi. Tällä tavoin vektoroitu aineisto viedään lopulta tietokantaan. Prosessissa voidaan vaikuttaa siihen, kuinka pienissä tai suurissa paloissa aineistoa käsitellään. Vektoroitua tietokantaa voidaan käyttää minkä tahansa kielimallin kanssa, mutta tietokannan vektoroinnissa käytetty upotemalli täytyy pysyä samana. Toisin sanoen, jokaista upotemallia kohden syntyy uusi vektoritietokanta.

Käyttöliittymä

Tekoälysovellukselle tuotettiin verkkoselaimella toimiva käyttöliittymä (WEBUI). Työkaluna käyttöliittymä toteutuksessa oli *Streamlit*. Tuotettu sovellus toimii yleisimmillä verkkoselaimilla (Edge, Chrome, Firefox). Pääsy sovellukseen on rajattu käyttäjätunnuksella ja salasanalla. Yhtäaikaisten käyttäjien määrää ei ole rajoitettu, mutta käytännössä moniajo hidastaa haun vasteiden saamista.

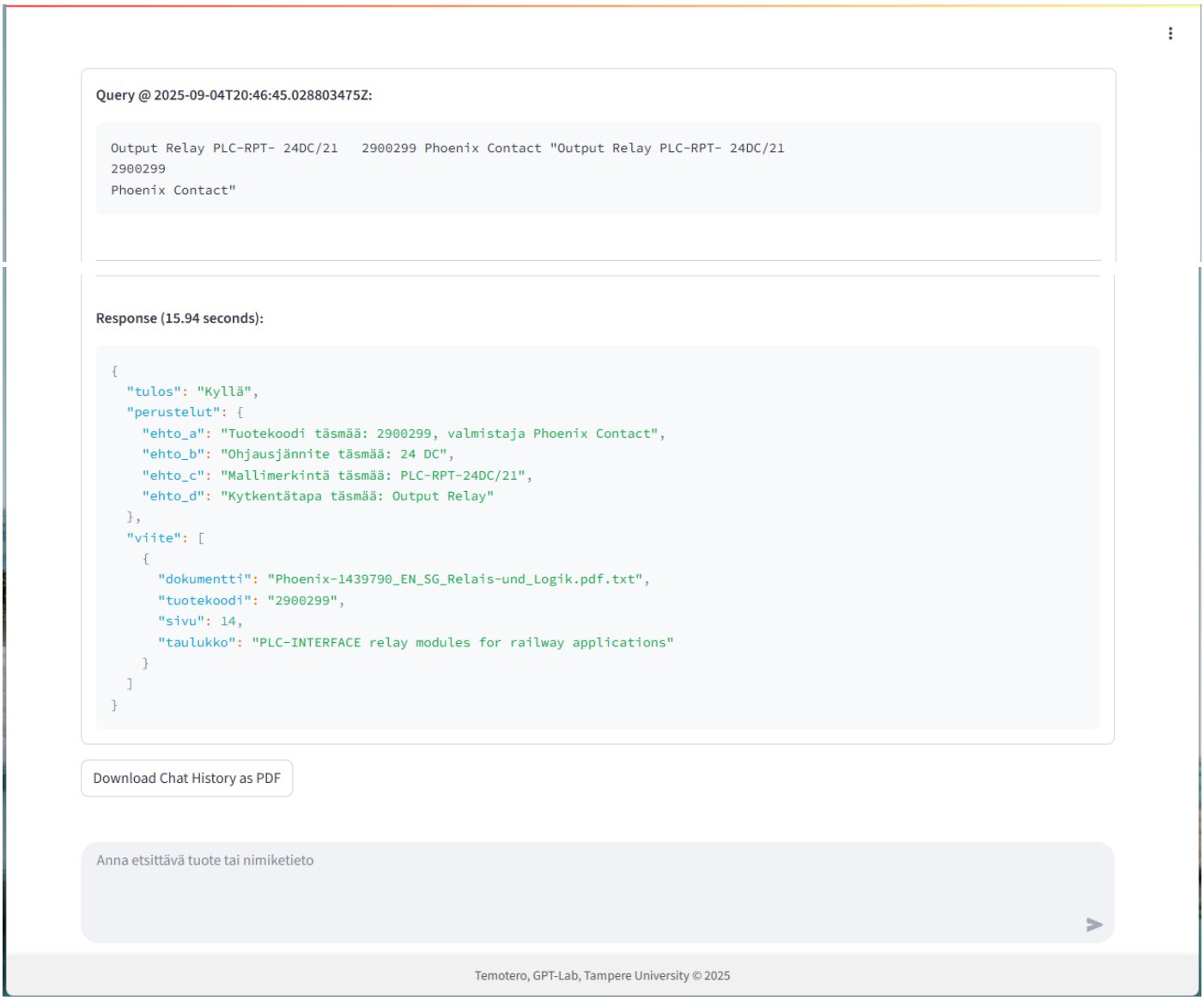
Näkymä sovelluksen aloitustilasta on esitetty seuraavassa kuvassa (kuva 3).



Kuva 3, Tekoälysovelluksen käyttöliittymän aloitusnäkymä

Sovelluksen aloitusnäkymässä kerrotaan sen käyttötarkoituksesta. Mukana on yksinkertainen ohje syötteestä ja sovelluksen vasteesta (tulos) sekä järjestelmätietoja kehitysversiosta ja käytetyistä kieli- ja upotemalleista.

Käyttäjän syöttämä haku syötetään sivun alalaidassa olevaan harmaaseen hakukenttään. Esimerkkihaku ja siitä saatu tulos on esitetty seuraavissa kuvissa (kuvat 4 ja 5).



Kuva 4 ja 5, Näkymä tekoälysovelluksesta saadusta hakuvasteesta

Haku koostaa seuraavista elementeistä:

1) Haku ja siihen annettu tekstisisältö (Query). Tässä on käytetty nimiketietoa lähteenä olleen taulukon neljästä kentästä.
2) Vasteaika (Response), jolla voidaan arvioida sovelluksen suorituskykyä ja sen yleistä toimintaa
3) Varsinainen vaste, joka koostuu kolmesta kohdasta:
    a. tulos (kyllä/ei)
    b. perustelut (ehtoja)
    c. viitetiedot (potentiaalisen nimikkeen sijainti tuoteluettelojen aineistoissa).

Käyttäjä voi tallettaa haun tulokset ulkoiseen PDF-tiedostoon.

Testaus ja pilotointi

Sovelluksen testejä on suoritettu sekä ulkoisessa palvelussa ChatGPT -modaalissa (käytössä versio *GPT-4o* sekä uusin versio *GPT-5*) sekä pilottia varten luodussa palvelinympäristössä. Yrityksen henkilöstön suorittamat pilotoinnit tehtiin vain paikallisella palvelinympäristössä. ChatGPT -modaalin käyttö tapahtui hanketiimin toimesta ja yliopiston käytössä olevalla lisenssillä, joka tarjoaa tutkimusryhmän käyttöön monipuoliset kielimallit, ilmaista käyttöä korkeammat käyttörajoitukset ja mm. suojaa aineiston tekoälyn kehitykseen liittyvältä opetusmateriaalikäytöltä.

Hanketiimin tekemien testien tuloksia

Tutkimusryhmän tekemiin testeihin valittiin kohteita kaikista kolmesta nimikedatojen pienryhmästä (6 kpl tiivisteitä, 7 kpl releitä ja 10 kpl ruuveja, yhteensä 23 testikohdetta).

Testaamista tehtiin samanaikaisesti sovelluksen tehtäväkuvauksen kehityksen kanssa, ja testausta tehtiin aluksi vain ulkoisessa palvelussa. Tehtäväkuvauksen sisällön kehittyessä ja palvelimen sisällön valmistuessa testauksen painotus siirtyi paikalliselle palvelimelle. Testejä on ajettu uudelleen tekoälyagentin kuvauksen kehittyessä. Testeistä 13 on arvioitu saaneen ok tuloksen ja 10 testiä on saanut NOK-tuloksen. Testien kokonaisuus on esitetty seuraavassa taulukossa (taulukko 5).

Taulukko 5, Tutkimusryhmän suorittamia testejä ja analyysejä niiden tuloksista

TIIVISTEET	Tulos (ok/nok)	Analyysi (vasteen rakenne, tulos, perustelut ja viite)
Tiivisteet # 1	OK	Vasteen rakenne on ok. Vasteen tulos on oikein, mutta perustelu ja viite yksityiskohdat puutteellisia
Tiivisteet # 2	NOK	Vasteen rakenne on ehjä, vasteen tulos on väärin, ja siitä johtuen myös viite on väärin (hallusinoitu)
Tiivisteet # 3	OK	Vasteen rakenne pirstoutuu. Vasteen tulos on oikein, perusteluissa on selitetty hakua ja viiteen tarkkuus on tyydyttävä
Tiivisteet # 4	OK	Vasteen rakenne on pääosin ok. Vasteen tulos on oikein, perustelut on selitetty ja viiteen tarkkuus on tyydyttävä
Tiivisteet # 5	NOK	Vasteen rakenne on pääosin ok. Vasteen tulos on epäselvä (ei voi sanoa, että se on oikein). Perustelut ovat hakutietoihin nähden kohtalaiset. Viitteitä on monia ja se heikentää tulosta
Tiivisteet # 6	OK	Vasteen rakenne on ok. Vasteen tulos on oikein, perustelut hyvät, viite rönsyää ja viite on tyydyttävällä tarkkuudella.
RELEET	Tulos	Analyysi
Releet # 1	NOK	Vasteen rakenne on ok. Vasteen tulos on väärin (etsittyä tuotekoodia ei ole aineistossa). Viitattu materiaali on lähin vastaava mutta tuotekoodi on kyselystä ja sivuviite tuotteiden yleisesittelyä.
Releet # 2	NOK	Vasteen rakenne on lähes ok. Vasteen tulos on väärin. Perustelut ja viite väärin.
Releet # 3	NOK	Vasteen rakenne on ok. Vasteen tulos on väärin. Perustelut ja viite väärin.

Releet # 4	NOK	Vasteen rakenne on ok. Vasteen tulos on väärin. Perustelut ja viite väärin.
Releet # 5	OK	Vasteen rakenne on ok. Vasteen tulos on oikein. Perustelut kunnossa. Viite osittain ok.
Releet # 6	NOK	Vasteen rakenne on ok. Vasteen tulos on väärin. Perustelut ja viite väärin.
Releet # 7	NOK	Vasteen rakenne on ok. Vasteen tulos on väärin. Perustelut ja viite väärin.
RUUVIT	Tulos	Analyysi
Ruuvit # 1	NOK	Vasteen rakenne on ok. Vasteen tulos on väärin. Perustelut ja viite väärin.
Ruuvit # 2	OK	Vasteen rakenne on ok, vasteen tulos vaikuttaa oikealta. Perusteluissa tuotetyyppi, koko (sisältyy taulukkoon) ja standardi oikein. Materiaalin osuus vaatii asiantuntijaa. Viitteessä dokumentti olisi oikein, mutta sivunumero väärin
Ruuvit # 3	OK	Vasteen rakenne on ok, vasteen tulos vaikuttaa oikealta. Perusteluissa tuotetyyppi, koko (sisältyy taulukkoon) ja standardi oikein. Myös materiaali ok. Viitteessä dokumentti olisi oikein, mutta sivunumero väärin (aika lähellä kuitenkin)
Ruuvit # 4	OK	Vasteen rakenne on enimmäkseen ok, vasteen tulos oikein. Perustelut osittain ok. Viitteessä dokumentti olisi oikein, mutta muut kohdat eivät.
Ruuvit # 5	OK	Vasteen rakenne on ok, vasteen tulos oikein. Perustelut enimmäkseen ok. Viitteessä dokumentti olisi oikein, mutta muut kohdat eivät.
Ruuvit # 6	OK	Vasteen rakenne on enimmäkseen ok, vasteen tulos oikein. Perustelut enimmäkseen ok. Viitteessä dokumentti olisi oikein, mutta muut kohdat eivät.
Ruuvit # 7	OK	Vasteen rakenne on ok, vasteen tulos oikein. Perustelut ok. Viite hyvälaatuinen.
Ruuvit # 8	OK	Vasteen rakenne on ok, vasteen tulos voisi olla ok. Perustelut ok. Viitteessä dokumentti olisi oikein, mutta muut kohdat eivät.
Ruuvit # 9	OK	Vasteen rakenne melkein ok, Vasteen tulos oikein, perustelut enimmäkseen ok. viittaukset tarpeettomia
Ruuvit # 10	NOK	Vasteen rakenne on ok. Vasteen tulos ja perustelu on väärin. Kyseessä oli synteettinen ja tiedoiltaan ristiriitainen nimike.

Kunkin testikohteen osalta on analysoitu, onko tekoälyagentin tuottama vaste käyttäjälleen hyödyllinen. Tärkein arviointikohde on vasteessa saatu **tulos** ja se, pitääkö se (arvioijan tietotaidon perusteella) paikkansa. Tämä on myös nimiketestin päätulos. Tuloksen jälkeen on tutkittu vasteessa annettuja **perusteluja,** joilla tekoälyagentti selvittää hakuehtojen täyttymistä. Ehtojen kirjausten tulee olla selkeitä. Tämän jälkeen on analysoitu vasteessa saadun viitteen laatua. Laadukas viite on sellainen, jossa on annettu yksittäinen kohdeaineisto ja siellä tuotekoodi, jonka avulla käyttäjä pääsee nopeasti suoraan haetun nimikkeen tuotetietoihin. Vasteen rakenne katsotaan kokonaisuutena. Laadukas vasteen rakenne on ehjä ja tiivis (kuten aiemmin esitetyssä kuvassa 5).

Tutkimusryhmän tekemistä testeistä on olemassa erillinen (ja tarkempi) dokumentti ”Nimikehakutestejä tekoälyllä”, ja sen uusin versio on päivitetty ja jaettu pilotin yhteyshenkilöille.

Dokumenttiin on koottu esimerkkejä nimikehaun kohteista tuoteryhmittäin. Kukin esimerkin alle on kirjattu haussa käytetty merkkijono sekä siitä saatu vaste (ja testissä käytetty tehtäväkuvauksen versio).

Yrityksen edustajien suorittaman pilotoinnin tuloksia

Yrityksen edustajien suorittamaan pilotointiin sisältyi useita eri nimikkeitä eri nimikeryhmistä. Testaajat saivat valita vapaasti kohteita kolmesta eri nimikeryhmästä.

Hanketiimille jaetuissa materiaaleissa oli testattu seuraavia kohteita (21 kpl): Tiivisteitä 7 kpl, releitä 13 kpl ja ruuveja 1 kpl. Testeistä 9 on arvioitu saaneen ok tuloksen ja 12 testiä on saanut NOK-tuloksen. Testien kokonaisuus on esitetty seuraavassa taulukossa (taulukko 6).

Taulukko 6, Yrityksen edustajien suorittamia testejä ja analyysejä niiden tuloksista

TIIVISTEET	Tulos (ok/nok)	Analyysi (vasteen rakenne, tulos, perustelut ja viite)
Tiivisteet # 1	NOK	Vasteen tulos on väärin. Tuotetta ei ole aineistossa
Tiivisteet # 2	NOK	Vasteen tulos on väärin. Tuotetta ei ole aineistossa. Perusteet ja viite hallusinoitu
Tiivisteet # 3	NOK	Vasteen tulos on väärin. Tuotetta ei ole aineistossa. Perusteet ja viite hallusinoitu
Tiivisteet # 4	OK	Vasteen tulo on (EI) on oikein ja vaste on perusteltu. Viite on kohtalaisesti täytetty
Tiivisteet # 5	OK	Vasteen tulo on (EI) on oikein ja vaste on perusteltu. Viite on täytetty huonosti
Tiivisteet # 6	NOK	Vasteen tulos on väärin. Tuotetta ei ole aineistossa. Perustelu ja viite on rakennettu kyselystä
Tiivisteet # 7	NOK	Vasteen tulos on väärin (perustelu pohjautuu väärään tuotteeseen, joten tulosta ei voi sanoa oikeaksi)
RELEET	Tulos	Analyysi
Releet # 1	NOK	Vasteen tulos on väärin. Tuotetta ei ole aineistossa. Perusteet ja viite kirjattu asiallisesti
Releet # 2	NOK	Vasteen tulos on väärin. Tuotetta ei ole aineistossa. Perusteet ja viite kirjattu asiallisesti
Releet # 3	NOK	Vasteen tulos on väärin. Tuotetta ei ole aineistossa. Perusteet ja viite hallusinoitu
Releet # 4	NOK	Vasteen tulos on väärin. Tuotetta ei ole aineistossa. Perusteet ja viite kirjattu asiallisesti
Releet # 5	OK	Vasteen tulos on oikein, tuote löytyy aineistosta, perustelut ja viite paranivat uudemmalla versiolla.
Releet # 6	OK	Vasteen tulos on oikein, tuote löytyy aineistosta, perustelut ja viite paranivat uudemmalla versiolla.
Releet # 7	OK	Aiempi vasteen tulos (ei) on uudemmalla versiolla korjaantunut (kyllä), ja perustelu sekä viite ovat selkeät
Releet # 8	OK	Aiempi vasteen tulos (ei) on uudemmalla versiolla korjaantunut (kyllä), ja perustelu sekä viite ovat selkeät
Releet # 9	NOK	Aiempi vasteen tulos (ei) on uudemmalla versiolla korjaantunut (kyllä). Perusteet ja viite hallusinoitu

Releet # 10	OK	Vasteen tulos on oikein, tuote löytyy aineistosta, perustelut osin arvailua ja viite keskeneräinen
Releet # 11	NOK	Vasteen tulos on väärin. Tuotetta ei ole aineistossa, perustelut ja viite on hallusinoitu
Releet # 12	NOK	Vasteen tulos on väärin. Tuotetta ei ole aineistossa, perustelut ja viite on hallusinoitu
Releet # 13	OK	Vasteen tulos on (osittain) oikein, osa tuoteryhmästä löytyy aineistosta, perustelut ja vasteen rakenne on ok
RUUVIT	Tulos	Analyysi
Ruuvit # 1	OK	Aiempi vasteen tulos (kyllä) on uudemmalla versiolla korjaantunut (ei), ja perustelu ovat selkeät

Testausaineisto koostui sovelluksesta PDF-tallennusta hyödyntäen otetuista tulosteista. Hanketiimin toimesta yrityksen tuottamat testit toistettiin uusimmalla tekoälyagentin kehitysversiolla ja uusimman vasteen sisältöä verrattiin yrityksen edustajien aineistoon kirjaamiin huomioihin. Taulukossa käytetyt tunnistetiedot on kirjattu testausaineiston varsinaisiin testiraportteihin jäljitettävyyden varmistamiseksi.

Analysointia ja jatkokehitysideoita

Tekoälyagentille luotu tehtävänkuvaus asetti agentin tutkimaan, onko käyttäjän kysymää tuotetta sovellukselle ladatuissa aineistoissa. Agentti sai etsiä vastaavuutta vain annetusta aineistosta. Agentin hakutehtävää varten oli määritetty joukko ehtoja, ja kaikkien niiden piti täyttyä, jotta haun tulos olisi haetun tuotteen kannalta positiivinen. Haun vasteessa agentin tehtävänä oli luoda viite, joka auttaa käyttäjää paikantamaan kyseisen nimikkeen nopeasti ja helposti oikeaan aineistoon ja sen oikeaan kohtaan. Pilotin loppupuolella ehtoja tarkennettiin vielä niin, että sähköteknisillä nimikkeillä (releet) oli omat hakuehtonsa ja vasteen rakenne verrattuna muihin tuoteryhmiin (tiivisteet ja ruuvit).

Tekoälyagentin toimintaa analysoitiin agentin tehtäväkuvauksen kehitysiteraatioiden sekä hanketiimin ja yrityksen tapaamisten (2 kpl) yhteydessä. Tapaamisten yhteydessä hanketiimi kirjasi ylös testattujen hakujen ja niiden tulosten osalta käytyjen keskustelujen huomiot ja havainnot. Ensimmäisen keskustelun osalta Tekoälyagentin toimintaa päivitettiin. Jälkimmäisessä tapaamisessa saadun palautteet on käsitelty suoraan analyysi ja kehitystoimina.

Tekoälyagentin toiminnan analysointia

Agentin toimintaa on analysoitu seuraavien kysymysten muodossa. Kysymykset on muotoiltu jälkikäteen tapaamisten muistioista koostuvien keskustelujen ja niiden yhteenvetojen pohjalta.

Pysyikö agentti sille osoitetussa aineistoissa?

Kyllä, agentti loi vasteen annetun aineiston pohjalta, ei sen oman mahdollisen koulutusdatan pohjalta. Testituloksissa ei näkynyt viittauksia aineiston ulkopuolisiin tietoihin. Tämä toimi sekä paikallisessa että kaupallisessa versiossa.

Oliko tekoälyllä käytössä riittävästi aineistoa?

Aineistoa olisi saanut olla enemmän. Aineiston kartoitukseen varattu aika oli rajallinen ja yksittäisen aineiston hankinta ja valmistelu arvioitua työläämpää. Aineiston haltijoiden lisäämät rajoitukset aineistoon pääsyssä ja sen lataamisessa kuluttivat työvaiheelle varattua aikaa.

Oliko käyttöliittymä riittävän helppo?

Kyllä. Varsinainen käyttö oli lopulta hyvin suoraviivaista. Ohjetekstiin olisi kaivattu mallisyötettä.

Oliko agentin toiminta riittävän nopeaa?

Kyllä. Vasteaikojen perusteella haut käsiteltiin enimmäkseen alle 30 sekunnin ajassa, joka on palvelimen suorituskykyyn nähden kohtuullinen tulos. Haun vasteen mennessä tuon aikarajan yli oli lopputuloksena virheellinen vaste tai agentin toiminnan rikkoutuminen (toimintahäiriö).

Pilotissa ei erikseen seurattu prosessin vasteajan ylittymistä ja osa pidemmistä vasteista voi johtua siitä, että kehitysympäristössä tehdään myös muita tekoälyyn liittyviä kokeiluja ja kehitystoimia.

Agentin tuottamaa vastetta ei saanut kokonaisuudessaan PDF-tiedostoksi. Puuttumaan jäi esitetty vasteen käsittelyaika. Näin ollen vasteaikojen etsintä jälkikäteen yrityksentuottamista tulosteista ei onnistu. Hanketiimin tekemissä testeissä vaste kopioitiin ruudulta erilliseen testausdokumenttiin, joten siellä vasteajatkin on kirjattu.

Vasteaikaa ei käytännössä ollut kokonaisuudessa mitenkään merkittävä. Sovelluksen sai jätettyä ajoon taustalle ja tuloksiin palattua myöhemmin. Paljon suurempi aika kului vastaan tulosten analysoinnissa, jossa käyttäjän piti selata tuoteluetteloaineistoja agentin antamiin viitteisiin liittyen. Myös yrityksen nimikeaineistoon liittyvät valinnan olivat välillä työläitä. Tämä olisi vaatinut hieman sisäisiä valmisteluja. Vertailuna hanketiimin testeissä nimitietokanta oli valmisteltu niin että hakuun valittavan tiedon sai kopioitua helposti. Testitulosten arkistointi tuotti myös omia lisätyötoimia. Nämä toimet liittyvät pilotointiin ja siihen kuluvaan työhön. Tässä tapauksessa pilotointi osoittautui ennakoitua työläämmäksi.

Noudattiko agentti annettuja ehtoja?

Kyllä ja ei. Vasteen ehtoihin liittyvät perusteet viestivät siitä, että agentti on käsitellyt kunkin ehdon kohdalla ehtoon liittyvää sisältöä, mutta se, miten agentti on tulkinnut ehdon sisällön, viestii usein tarpeesta saada täsmällisempi ohjeita. Agentin toimintaan liittyviä epäselvyyksiä on analysoitu kaupallisen ChatGPT -version kanssa käydyissä ”keskusteluissa”, ja sieltä saatujen tulosten myötä ohjeita on pyritty selkeyttämään. Ongelmallista kehittäjän kannalta on se, että vaikka ChatGPT olisi keskustelun päätteessä arvioinut ohjeen sisällön toimivaksi ja korjatuksi, saattoi sekin seuraavassa testissä todeta ohjeen olleen edelleen epäselvä.

Noudattiko agentti ohjeita viitteen rakentamisessa?

Kyllä ja ei. Agentille ohjeistettu vaste oli suunniteltu koneluettavaan muotoon, jossa on selkeä ja yksinkertainen rakenne. Agentti kyllä rakensi vasteensa annetun mallin suuntaan. Useissa testeissä ilmeni, että agentti laajensi vastetta generoimalla vastaavaa rakennetta pidemmäksi. Tämä asettaa haasteita käsitellä agentin vasteita koneellisesti.

Minkälaisia ongelmia agentin toimissa ilmeni?

Yleisesti ottaen, tarkka nimikehaku vaikuttaa olevan hankala tehtävä luovalle tekoälylle ja vektoritietokannan käytölle. Siinä missä käyttäjän tekemät kyselyt vaikkapa pöytäkirjojen sisältämistä vastauksista tiettyyn teemaan tai aiheeseen ovat vastaavissa piloteissa toimineet hyvin, näyttää teknisen tuotetiedon etsintä haastavalta toiminnolta. Pilotissa käytössä olleen pienen kielimallin osalta tämä voisi olla jopa oletettu tulos mutta vastaavat ongelmat vaivasivat yhtä lailla kaupallista ”huipputuotetta”.

(suorat vastaavuudet tuotekoodin tai valmistajan mukaan)

Suurin haaste vaikutti liittyvän kyselyssä annetun tuotekoodin, valmistajatiedon (tai sitä vastaavan tunnistetiedon) aiheuttama hämmennys agentille. Näiden tietojen mukanaolo kyselyssä johti positiiviseen tulokseen, vaikka tuotekoodia tai valmistajaa ei ollut etsityssä aineistoissa. Tämä heijastui myös vasteessa, jossa agentti näytti ottavan tuotekoodin vasteeseen suoraan käyttäjän antamasta kyselystä. Haun kannalta annetun tuotekoodin ja valmistajatiedon suora vastaavuus aineistoon on tärkeää, mutta pilotissa suunnittelun agentin toiminnalle liian monimutkainen. Lisähuomiona tekoälyn kyky ”suodattaa” pieniä kirjoitusvirheitä kääntyy tässä tehtävässä sitä vastaan. Tekoäly osasi tulkita nimekehaussa virheellisesti kirjatun valmistajanimen oikeaksi, mutta laajemmassa käytössä tämä voisi johtaa ongelmiin lähellä toisiaan olevien nimien kesken. Tämä avaan myös huolen siitä, kuinka hyvin tekoäly lopulta osaa erottaa hyvin lähellä toisiaan olevia nimikkeitä.

(mitta- ja kokotiedot sekä muut ominaisuudet)

Agentille nimikehaun tai aineiston kokotiedoissa olevat mittatiedot ja niiden desimaalit eivät eroa sanallisen lauseen välilauseiden pilkuista tai pisteistä. Tämä korostuu vektoritietokannan haussa, jossa yksittäiset merkkijonojen vastaavuudet merkitsevät enemmän kuin ihmisen käsite numeroista. Lisäksi valmistajien tavat kirjata koko- ja mittatietoja vaihtelevat. Jos yksittäiset mittatiedot ovat haastavia, niin taulukot kokoväleistä (pultit) ovat vielä vaikeampia. Agentille luotiin ohjeistusta, jossa haettua mittaa tulisi verrata valmistajan antamaan mittaväliin. Vaikka agentti näin näytti toimivan, oli sillä vaikeuksia saada vertailu tehtyä tietyn mittataulukon sisällöstä. Testissä ilmeni myös yksittäistapauksia, joissa nimiketietoihin oli kirjattu ”vaihtoehtoinen” koko. Tämä virheellinen kirjaustapa huomattiin vasta keskusteluissa ja agentin ohjeeseen sisällytetty tuki koko vaihtoehdoista otettiin pois. Ongelmaksi jää se, että nimihaussa voi olla ristiriitaisia tietoja, eikä agentti voi tietää mikä on oikea hakukriteeri. Usein nimiketietoon sisältyy teknisiä lisätietoja, joiden tunnistaminen tuotti haasteita. Standardit toimivat kohtalaisen hyvin, mutta materiaalin pinnoitteet yms. olivat vaikeita.

Agentilla oli ongelmia erottaa sähköteknisten tuotteiden käsittelyyn annettuja erikoisohjeita. Pilotin loppupuolella lisättiin ominaisuus, jossa tietyt sähkötekniset tuotteet erotetaan omaksi käsittelypoluksi ja se vaikutti toimivan erittäin hyvin. Tätä näkyy vasteiden rakenteessa, jossa

sähköteknisille tuotteille on erilaiset vertailukohdat. Agentin kielimallin ja työohjeen pieni koko asettavat rajoituksia erottelulle.

Perustietojen lisäksi kyselyissä oli mukana paljon muutakin tietoa ja agentilla oli haasteita tunnistaa, mitä mikäkin tieto tarkoittaa. Osassa vasteita agentti on voinut hienosti selittää, miten haussa mukana ollut materiaalikoodi muunnetaan tuoteluettelon vaihtoehtoihin sopivaksi, tai kuinka vanhalla standardinimellä tehty haku vastaa uudelle standardilla luokiteltuja nimikkeitä. Välillä agentti on voinut sekoittaa vakavalla tavalla sähkötekniset jännitteet (230 AC ja 24 DC).

(vasteen sisällön hallusinaatiot)

Vasteen sivunumeroinnissa esiintyi usein ns. *hallusinaatiota*. Tämä liittyy tekoälyn toimintatapaan, jossa se pyrkii tekemään tehtävänsä loppuun, vaikka sen aineiston sisällön laatu olisi liian huonoa. Tässä tarkoittaa sitä, että sivunumeroon liittyvä matemaattinen yhteys sisältöön on lähellä nollaa mutta prosessina sekin kelpaa. Tämä voi viitata parsintaprosessin sisällä olevaan käsittelyjärjestykseen (johon agentin ohjeistus ei ota kantaa).

(vasteessa käytetty kieli ja muita yksittäisiä ilmiöitä)

Agentin työohjeessa vasteen kieleksi määritettiin Suomi. Tästä huolimatta agentti saattoi joskus vastata englanniksi, varsinkin jos nimikehaussa tai aineistossa oli paljon englanninkielisiä termejä. Kielimallin tuki useammalle kielelle on yleensä hyvä, koska keskustelua eri kielistä aineistosta voidaan käydä käyttäjän omalla kielellä. Joskus kielituki näyttäytyy nurjasti vasteiden väärällä kielellä.

Onko agentin toiminta riittävän laadukasta?

Testien perusteella todettiin, että agentin toiminta vaatii runsaasti jatkokehitystä ja vasteiden tarkistaminen vielä ihmisasiantuntijan läsnäoloa.

Tekoälyn kyky tuottaa vakuuttavan oloista tekstiä hämmensi käyttäjiä. Vasteessa voitiin saada positiivinen tulos ja varsin kattavilta vaikuttaneet perustelut ehtojen suhteen. Kuitenkin tarkistus aineistosta tuotti negatiivisen lopputuleman. Hyvältä näyttävä tulos ei siis ole tae prosessin laadusta.

Pilotin aikana vertailtiin paikallista kielimallia sekä kaupallista palvelutarjontaa (hanketiimillä ChatGPT ja yrityksellä Copilot). Kaupallisten versioiden etuja on niiden suurempi kielimalli, mutta siitä huolimatta niidenkin kyvykkyys jäi tässä tehtävässä heikoksi.

Pilotissa saatujen kokemusten perusteella tekoälyagentti ei voi korvata ihmisasiantuntijaa. Agentti voi parhaillaan toimia väsymättömänä assistenttina, joka kahlaa sille osoitettua aineistoa ja tuottaa hakujen perusteella suosituksia asiantuntijalle. Asiantuntija voi nopeasti analysoida ja tarkistaa suosituksen tuloksen.

Tekoälyagentin toiminnan jatkokehittäminen

Tekoälyagentin kehitys ja ylläpito on jatkuvaa työtä. Tässä pilotissa on tuotu esiin, mitä tarpeita tekoälyagentin käyttö tuo palvelinresursseille ja sovelluksen käytölle sekä varsinaisen agentin sisällön kehittämiseen. Pilotin alussa tehty materiaalien kartoitus on hyvä esimerkki aineistojen hankintaan ja valmisteluun liittyvästä työstä.

Tekoälyagentin tehtävänkuvaukseen määritetty kuvaus on juuri niin hyvä kuin on määritystä luoneen henkilön tai tiimin käsitys aiheesta. Pienen kielimallin tarve yksinkertaisesta ja tiiviistä kuvauksesta on haastava tehtävä suunnittelijalle. Kaupallisella suurella kielimallilla on parempi kyvykkyys käsitellä laajempaa työohjetta, mutta silläkin on rajansa ohjeen koosta. Lisäksi ohjeen laajeneminen kasvattaa riskiä ohjeen sisällössä olevista epäkohdista ja puutteista.

Pilotin kohteena ollut laaja nimiketietokanta on kohteena hyvin haastava, sillä sisältö uudistuu jatkuvasti ja sitä kautta myös tekoälyagentin toimia tulee varautua uudistamaan.

Tekoälyn toimintaa ei voi suoraan verrata perinteiseen ohjelmistoon, sen suunnitteluun ja siinä käytettäviin sääntöpohjaisiin etenemispolkuihin. Tekoälyn etu ohjelmistokoodiin verrattuna on siinä, että tekoäly saa itse etsiä parhaaksi näkemänsä reitin lopputuloksen saamiseksi ja sen toimintaa voidaan kehittää ylemmällä tasolla. Täsmällisen tiedon käsittelyssä tämä vapaus ei testien perusteella toimi kovin hyvin, koska prosessin kulku on paljolti ennalta määritettyä. Toiminnan yksityiskohtien lisääminen taas lisää ja siirtää kehityksen ja ylläpidon työtä. Ratkaisu vaatinee kuitenkin jonkinlaisen kompromissin.

(nimikeryhmiin erikoistuvia agentteja)

Toimintaa voisi kehittää ohjeiden ja tekoälyagenttien erottamiseen nimikekohtaisesti. Mitä kapeamman ja erikoistuneemman haun agentti saa suorittaa, sen selkeämpää on myös sen työn tulos. Myös aineisto voidaan osittaa agenttikohtaisiin tarpeisiin ja tällä tavoin parantaa upotemallin tulosten laatua. Useiden agenttien hallinta ja käyttö lisäävät prosessin monimutkaisuutta ja palvelinresurssien tarvetta. Aineiston pilkkominen taasen lisää luokittelu ja ryhmittelytyötä. Molemmat lisäävät kehitykseen ja ylläpitämiseen liittyvää työtä.

(työprosessin osittaminen)

Kehitystoimia voisi tehdä myös hakuprosessin osittamiseen. Myös tämä selkeyttäisi tekoälyagentin tehtäväkenttää ja mahdollistaa paremmin pienen kielimallin hyödyntämisen. Esimerkiksi nimikeryhmäkohtaisten agenttien käyttö edellyttää käytännössä myös sitä, että ensin haettava nimike on tunnistettava. Vastaavasti sähköteknisten tuotteiden nimikehaussa annettu valmistajatieto ja tuotekoodi, jota pitää noudattaa, tulisi ensin tunnistaa ja sitten hyödyntää rajaamalla vektorikannan hakua vain kyseisen valmistajatiedon omaaviin aineistoihin. Myös ei-sähköteknisten nimikkeiden osalta etsittävä nimike ja sen ominaisuudet tulisi ensin tunnistaa ja vasta sitten sille voidaan alkaa tehdä vastaavuuden hakua. Haussa annetut tekniset ominaisuudet pitäisi ensin saada määriteltyä yksittäisiin tietokomponentteihin kuten standardi, pituus, halkaisija ja materiaali.

Haun lopussa voitaisiin hyödyntää agenttia ja uudelleentarkistusta, jossa vasteessa saatu tulos ajetaankin ensin läpi tarkastavalla agentilla, jonka tehtävänä on todentaa vasteessa saatu viittaus. Jos tarkastava agentti onnistuu kohteen löytämisessä, voidaan vaste antaa eteenpäin käyttäjälle.

(tuoteluetteloaineistojen hankintaa)

Toimittajien ja valmistajien tuottamia (tuoteluettelot) aineistoja tulee kartoittaa aktiivisesti. Yksittäisen aineiston saaminen voi edellyttää runsaasti toimia. Isona toimijana yritys voisi tehdä vaikuttamistoimia tuoteluetteloiden tuottajien suuntaan, jotta nämä sujuvoittaisivat aineistojensa saantia. Aineistojen hankinnassa tulee ottaa huomioon julkaisijan määrittelemät käyttöluvat ja mahdolliset rajoitteet, näin erityisesti kaupallisissa käyttötarkoituksissa. Toisin sanoen, materiaalin maksuton latausmahdollisuus

ei välttämättä mahdollista sen käyttämistä automaattisessa/teknisessä prosessoinnissa ja hyödyntämisessä.

(tuoteluettelo- ja nimikeaineistojen esikäsittelyä)

Toimittajiin ja valmistajiin liittyviä aineistoja (tuoteluettelot) tulisi valmistella paremmin koneluettavaan muotoon. Esimerkiksi mittoihin liittyvät kirjaustavat voisi ”puhtaaksikirjoittaa” koneelle helpommin luettavaan muotoon. Myös taulukot kokoväleistä (kuten pultit ja ruuvit) voisi tuottaa yksittäisiksi tietoriveiksi. Muita esikäsittelytoimia voisivat olla tarpeettomien sisältöjen poisto tai kuvamuodossa olevien sisältöjen muunnos tekstiksi. Myös sisältöjen kääntäminen käyttökieltä vastaavaksi voisi selkeyttää toimintaa.

Yrityksen käytössä olevaa nimikeaineisto tulisi kehittää, jotta se sopisi paremmin koneluettavaan muotoon. Keskustelujen yhteydessä tuli esiin useita esimerkkejä tuotetiedon siivoamisen tarpeesta. Esimerkiksi kirjausvirheet nimissä tai mittatiedoissa tai haun kannalta tarpeettomat sisäiset viitetiedot eivät haittaa kokenutta ihmiskäyttäjää, mutta haittaavat tekoälyn toimia. Tekoälyn käyttöä varten voisi olla oma tietokenttä, jonne nimiketiedoista kootaan vain tekoälyn tehtävää varten tarvittavat tiedot.

(tuoteluetteloiden järjestelyä vektoritietokannassa)

Vektoritietokannassa olevia aineistoja voidaan osittaa valmistaja ja/tai nimikeryhmäkohtaisiin ryhmiin. Näin yksittäisen vektoritietokannan käyttö tuottaisi parempia tuloksia verrattuna siihen, että samassa aineistossa olisi mukana kaikkien valmistajien tuoteluetteloita.

(terminologiat tutuksi)

Agentin kyky tunnistaa ja ymmärtää teknisiä termejä liittyy sen käytössä olevaan opetusdataan. Pienen kielimallin kyvykkyys on hyvin rajattua, eikä kaupallinen versiokaan takaa, että jokainen nyanssi osataan tulkita oikein. Agentin toiminnan tueksi tulisi laatia tukiaineistoja. Esimerkiksi valmistajien nimet, standardit, pintakäsittelykoodit yms. voidaan tuottaa listoiksi, jotka voidaan sisällyttää upotemalleihin. Aineistoja hyödyntämällä agentti voisi tunnistaa haussa annetut tiedot niiden oikean merkityksen mukaisiksi.

(hallusinaatioiden torjunta)

Prosessin osittaminen ja jakaminen selkeämmiksi tehtävän osiksi sekä kielimallin opetusdataa tukevat lisäaineistot auttavat osaltaan toiminnassa ilmenevien hallusinaatioiden kitkemisessä. Myös kielimallien ohjauksessa käytettävät parametrit on hyvä tunnistaa ja hyödyntää.

(tehokkaammat resurssit)

Palvelimissa käytetyt GPU-yksiköt ja niiden kapasiteetti antavat rajoituksia käytettäville kielimalleille. Jos kielimallin koko (muistivaatimus) kasvaa, tulee myös palvelimien laskentakapasiteettia päivittää. Jos hakuja tehdään suuria määriä ja automatisoituna, tulee yksittäisten hakujen kestot ja palvelimen kapasiteetti huomioida. Kaupallisissa palveluissa käytöstä syntyvä laskentatarve heijastuu esimerkiksi rajapinnan käyttömaksuina.

Pienen kielimalin kyvykkyyteen liittyy rajoituksia. Näitä voidaan kiertää edellä esiintuoduin ehdotuksin prosessin osittamisesta. Tällöin samaa pientä kielimallia joudutaan kutsumaan useammin, mutta prosessi kokonaisuutena saadaan mahdutettua paikalliselle palvelimelle.

Pilotissa on osoitettu, ettei tekoäly-ympäristön hankinta ole investointina kovin suuri, ja GPU-yksiköihin liittyvä kehitystyö tuo markkinoille entistä kustannustehokkaampia ratkaisuja. Yrityksen tulee kuitenkin miettiä tekoäly-ympäristön sopivaa sijoituspaikkaa huolella ja hyvissä ajoin.

Kielimallien sisältöön ei useinkaan ole näkyvyyttä. Kielimallin vaihtamiseen tai päivittämiseen liittyy omat riskit siitä, että jokin muutos on sellainen, että se muuttaa agentin toimintaa merkittävästi. Tätä riskiä voinee pienentää esimerkiksi tuottamalla vektoritietokantaan tukimateriaalia kielimallin tueksi. Kokonaan oman kielimallin kehittäminen on sekin mahdollista, mutta siihen tarvittavien investointien ja saatujen hyötyjen suhteesta on syytä laatia erillinen selvitys.

(erillinen AI-kehittäjätiimi)

Tekoälyn ja agentin kehittäminen on oma asiantuntijuuden alansa. Toisaalta tekoälyn kehittäjältä ei pidä vaatia syvällistä nimikkeisiin liittyvää asiantuntijaosaamista. Yhteistyössä nämä asiantuntijat voivat kehittää molempia puolia. Koneluettavuuden tarve tuo näkemystä nimiketiedon kehittämiseen ja nimiketietoon liittyvät näkemykset auttavat kehittäjää tunnistamaan tekoälyn ja agentin käyttöön liittyviä ongelmia.

Yritykselle tämä tarkoittaa investointeja tekoälyn käyttöön sekä tarvittavien laiteresurssien mutta ennen kaikkea AI-asiantuntijuuden kehittämiseen. Tekoälyn hyödyntämiseen liittyvät yleiskoulutukset auttavat henkilöstä ymmärtämään tekoälyn roolin ja kyvykkyyden, mutta varsinainen yrityksen erityistarpeisiin liittyvä kehitysosaaminen syntyy vain yrityksessä tehdyn kehitystyön kautta.

(toiminnan automatisointi)

Keskusteluissa tuotiin esiin, että automatisoinnilla tavoitellaan kilpailutustoimissa kuluvien resurssien siirtoa etsintätyöstä kilpailutuksen toteuttamiseen.

Tekoälyprosessi on itsesään merkittävä automaation tason nosto nimiketietojen haun (osa kilpailutusprosessia) käsittelyssä. Tekoäly voi käsitellä valtavia määriä aineistoa yksittäisen haun aikana. Tämän kokoluokan työn tuottaminen muulla tavoin ei nykyhetkessä ole mahdollista.

Se, että tekoälyn kanssa toimimista automatisoidaan (esim. kaikkien nimikkeiden hakeminen kerralla), on järkevää vasta siinä vaiheessa, kun tekoälyn sisällä tapahtuva prosessi saavuttaa riittävän laadukkaan tason.

Pilotointiin ja yhteistyöhön liittyvät huomiot

Tässä osiossa on esitetty tiivistettynä pilotin palautekeskustelussa esiin nostettuja huomioita pilotista ja yhteistyön sisällöstä.

Jälkikäteen voidaan todeta, että pilotoinnin kohde osoittautui ennakoitua laajemmaksi työksi. Yhden agentin kyvykkyys kärsi liian laajasta sisällöstä. Sisältöä olisi voitu rajata pilotin aikana enemmän. Toisaalta liian voimakkaat rajaukset voivat estää tekoälyn kyvykkyyden esille pääsyn.

Agentin kehitystyötä olisi voitu käydä läpi nykyistä tarkemmin. Kehitystyöhön olisi voitu kutsua enemmän väkeä yrityksen puolelta. Ihmisiä olisi voinut kutsua aiemmin. Työryhmiä olisi voitu perustaa ja jakaa kehitystyötä osiin (kuten kehitysympäristö, agentti ja aineistot). Tätä kautta olisi saatu paremmin nimiketietoihin liittyviä kriteerejä selvitettyä. Testausta ja niiden palautteiden käsittelyä olisi

saanut olla enemmän. Lomakausi kulutti osan ajasta. Agentin ja sen vasteiden kanssa työskentely opetti, millaisia asioita agentilta voi kysyä. Tiiviimmille tutkija-asiantuntija-keskusteluille olisi ollut tarvetta.

Pilotin aikana saadut kokemukset ja näkemykset todettiin paljon syvällisemmiksi kuin usein tarjolla olevat päiväkohtaiset koulutukset. Pilotin aikana käytetty aineisto oli hyvin saatavilla ja asioista kerrottiin riittävällä tarkkuudella. Tarvittavan tiedon ja ymmärryksen tarpeen ymmärrys kehittyi pilotin aikana yrityksen edustajien osalta ja he kokivat, että olisivat halunneet olla enemmän läsnä ja mukana. Yrityksen edustajia olisi saanut ohjata enemmän. Raportointi oli laadukasta ja pilotilla on ollut aktiivinen ote.

Kehityskohteen valinta ei kaduttanut. Toiminta ja agentin mukaan saaminen liittyy jokapäiväiseen työhön. Muissa vaihtoehdoissa on myös potentiaalia, mutta pilotti on osoittanut, että myös niissä olisi ollut suuri työ tehtäväksi.

Kaikkea enemmän ja kauemmin? Palaute kuvastanee, että yhteistyö on ollut mielekästä ja hedelmällistä? Toisaalta toiveiden toteuttaminen olisi vaatinut paljon yksityiskohtaisemman ja pidemmän aikataulutuksen, enemmän resursseja hanketiimiltä eikä paketti kasvaessaan suuremmaksi olisi enää ollut hankkeen luonteen mukaista.

Pilotointiin varatun lyhyen aikataulun kannalta kehityskohde oli erittäin haastava ja monissa keskustelujen kohdissa korostui pidemmän ja laajemman yhteistyön tarve. Pilotin myötä yritys on saanut paljon arvokasta tietoa siitä kuinka paljon enemmän kehitystyöhön pitäisi varata resursseja.

Yhteystiedot

Lisätietoa tutkimusryhmästä ja hankkeesta sekä yhteyshenkilöt löytyvät seuraavasta taulukosta (taulukko 7).

Taulukko 7, Lisätietoja tutkimusryhmästä ja hankkeesta

Organisaatio	Yhteyshenkilö ja verkkosivut
GPT-Lab-tutkimusryhmä	Tutkimusryhmän johtaja Professori Pekka Abrahamsson, Tampereen yliopisto (Pori) Sähköposti: pekka.abrahamsson@tuni.fi puhelin: +358405415929 profiili: https://www.tuni.fi/fi/ihmiset/pekka-abrahamsson https://www.tuni.fi/fi/tutkimus/gpt-lab https://gpt-lab.eu/
Teknologiayritysten modernisointi tekoälyn ja robotiikan avulla (temotero) -hanke	Hankkeen projektipäällikkö Janne Harjamäki, Tampereen yliopisto (Pori) sähköposti: janne.harjamaki@tuni.fi puhelin: +358408262714 profiili: https://www.tuni.fi/fi/janne-harjamaki https://www.tuni.fi/fi/tutkimus/teknologiayritysten-modernisointi-tekoalyn-ja-robotiikan-avulla-temotero https://www.avoinsatakunta.fi/temotero/ https://www.roboai.fi/tutkimus-ja-tuotekehitys/projektit/temotero/

LIITTEET

LIITE A (tekoälyagentin tehtäväkuvaus)

Tekoälyagentille määritetty tehtävänanto (osana sovellukselle annetuista parametreista). Julkisversiossa rajapinta-avain, verkko-osoitteet ja tehtävänannon yksityiskohdat on jätetty pois.

```
VERSION="1.0.0"
# Assumes Ollama is available somewhere
OPENAI_API_KEY=xxxx
OLLAMA_BASE_URL=xxxx
OLLAMA_BASE_URL_EMBED=xxxx
EMBEDDING_MODEL=snowflake-arctic-embed2
#EMBEDDING_MODEL=bge-m3
#EMBEDDING_MODEL=llama3.2
#LLM_MODEL=hf.co/mradermacher/Llama-Poro-2-70B-Instruct-i1-GGUF:Q4_K_S
LLM_MODEL=hf.co/mradermacher/Llama-Poro-2-8B-Instruct-GGUF:Q4_K_M
#LLM_MODEL=hf.co/LumiOpen/Poro-34B-chat-GGUF:latest
#LLM_MODEL=llama3.3
#LLM_MODEL=llama3.3:70b-instruct-q2_K
#LLM_MODEL=llama3.2

# Various configurations
CHUNK_SIZE_TOKENS=128
FILE_SUFFIX=".txt"
# Finetune and validate Ollama settings
OLLAMA_LLM_OPTIONS = '{
   "num_ctx": 16384,
   "num_predict": 512,
   "temperature": 0.5
}'

#    "max_tokens": 512,
#    "temperature": 0.6,
#    "top_p": 0.9,
#    "frequency_penalty": 0,
#    "presence_penalty": 0,
#    "stop": []
# }'
EXAMPLE_QUERY = 'xxxx'

# Assistant related prompt construction
ASSISTANT_INSTRUCTIONS='{task}

{structure}'
ASSISTANT_INSTRUCTIONS_TASK='# TEHTÄVÄ
xxxx

# PÄÄTÖS
xxxx

# EHDOT
xxxx

# HUOMAUTUS

xxxx
'

ASSISTANT_INSTRUCTIONS_STRUCTURE='# Rakenne
```

Konteksti
Kontekstin rakenne on JSON-muotoinen, ja se syötetään käyttäjän kysymyksen ohessa:
```json
{ "context": [ { "source": "source document name  chunk_n-of-m", "similarity": 1.0, "chunk": "content from the source document" } ] }
```

Kysymys
Käyttäjän kysymyksen rakenne on seuraava:
```plaintext
query: example question
```

Vastaus
Vaste palautetaan AINA Suomen kielellä.

Vaste palautetaan AINA JSON-muodossa seuraavalla rakenteella:
xxx

Esimerkki 1: xxx

```xxx
```
Esimerkki 2: xxx

```xxx
```

Esimerkki 3: xxx

```xxx
```
Esimerkki 4: xxx

```xxx
```
'
ASSISTANT_CONTEXT_PATTERN="{context}"
ASSISTANT_QUERY_STOPWORD="query"
ASSISTANT_RESPONSE_STOPWORD="response"
ASSISTANT_PROMPT_PATTERN="```plaintext
{query}
```

```json
{context}
```

{response}"
```