

TARTU ÜLIKOO
Arvutiteaduse instituut
Andmeteaduse õppekava

Karl-Erik Kanal

**Genoomiotsingu töövoogude arendamine
BioCC OÜ jaoks
Magistritöö (15 EAP)**

Juhendaja: Martin Rebane, PhD

Tartu 2025

Genoomiotsingu töövoogude arendamine BioCC OÜ jaoks

Lühikokkuvõte:

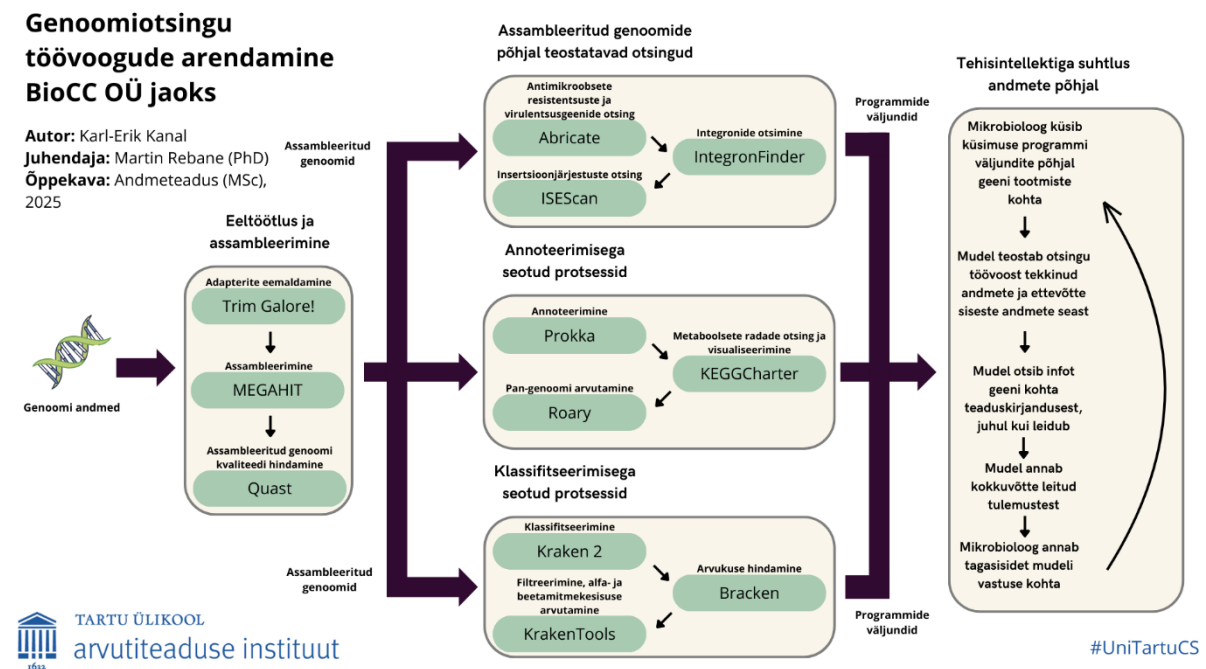
Ettevõtte BioCC OÜ tootmisprotsessis on läbivalt vajalik mikroorganismide genoomi ja ainevahetusproduktide, ehk metaboliitide uurimine. See rutiinne protsess aga võtab tarbetult kaua aega, mida BioCC töötajad võiksid kasutada muude sisukamate ülesannete jaoks. Töö eesmärk on luua töötajaid assisteerivad töövood, mis automatiseeriksid enamuse sellest protsessist ning seeläbi vähendaksid töötajate ajakulu ning looksid eeldused andmete lisaväärindamiseks. Töö raames disainitakse need töövood koostöös ettevõttega. Esialgsetest mõtetest luuakse nende jaoks toimiv lahendus. Kokku valmib kaks automatiseeritud töövoogu ning nendega kaasnev AI agent, mis ühendab omavahel teaduskirjanduse ning geenitootmisandmed. Lõpus analüüsitakse valminud tööd ning tuuakse välja, milliseid edasiarendusi oleks tulevikus nende puhul võimalik teha.

Võtmesõnad:

Mikrobioloogia, metaboliidid, genoomiotsing, tehisintellekt, töövoog, n8n, AI agent

CERCS: P175 Informaatika, süsteemiteooria; P176 Tehisintellekt

Visuaalne kokkuvõte:



The development of genome search workflows for BioCC OÜ

Abstract:

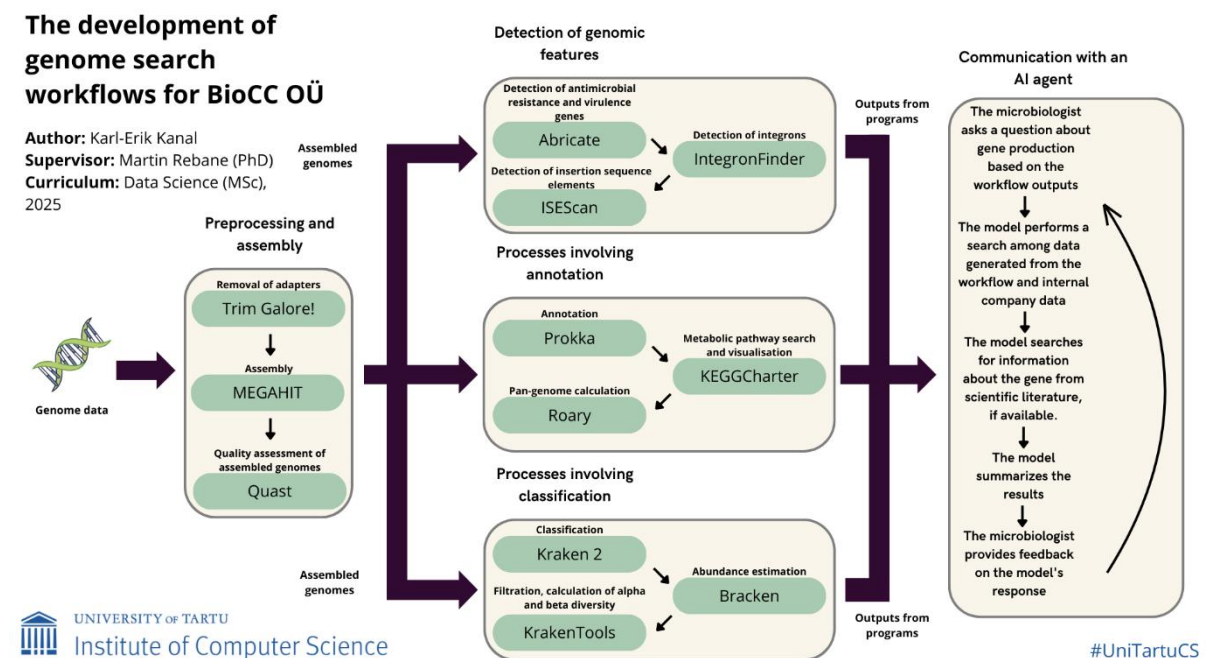
The company BioCC OÜ consistently requires the study of microbial genes and their metabolic by-products, also known as metabolites, in its production process. This routine process, however, takes up a lot of time, which could be better used for other tasks. The aim of this work is to create assisting workflows for the workers, which would automate most of this process. As part of this work, the workflows were developed in collaboration with the company, transforming initial ideas into a working solution. In total, two automated workflows are created along with an accompanying AI agent, which joins together both production data and information from scientific literature. Finally, the completed workflows and agent are analysed and potential future improvements are proposed.

Keywords:

Microbiology, metabolites, genome search, artificial intelligence, workflow, n8n, AI agents

CERCS: P175 Informatics, systems theory; P176 Artificial Intelligence

Visual abstract:



Sisukord

Tänuavaldused.....	6
Sissejuhatus	7
1 Mõisted, terminid ja kasutatavad lühendid	9
2 Tehisaru potentsiaal genoomiotsingul kasutamiseks	10
2.1 Allikapõhine genereerimine ehk RAG	10
2.2 Iseseisev tööriistade kasutamine	11
2.3 Agentne arutlemine ja mitmikagentsüsteemid	12
3 Genoomiotsingu töövoogude arendusprotsess	15
3.1 Iteratiivne lahendusprotsess	15
3.2 Nõuete täpsustamine	15
4 Peamise töövoog ja alamtöövoog ülesehitus.....	17
4.1 Eeltöötlus ja assambleerimine	17
4.1.1 Adapterite eemaldamine	18
4.1.2 Assambleerimine	18
4.1.3 Kvaliteedihinnangu andmine.....	18
4.2 Assambleeritud genoomide põhjal teostatavad otsingud	19
4.2.1 Antimikroobse resistentsuse ja virulentsuse geenide otsing	20
4.2.2 Integronide otsing	20
4.2.3 Insertioonjärjestuste otsing	20
4.3 Genoomide klassifitseerimisega seotud analüüsid	21
4.4 Genoomide annoteerimisega seotud analüüsid	22
5 Tulemused	24
5.1 Töövoogude kokkupanekuks kasutatud tehnoloogiad	24
5.1.1 Keskkond n8n.....	24
5.1.2 Andmebaas	24
5.2 Peamine töövoog	25

5.3	Alamtöövoog.....	27
5.4	Suhtlemise agent	28
5.4.1	Andmebaasist otsimise tööriist.....	29
5.4.2	Teadusartiklite otsimise tööriist	32
5.4.3	Dokumentidest otsimise tööriist.....	33
5.5	Failide üles laadimise voog	36
5.6	Võimalikud edasiarendused	36
5.7	Tulemuste valideerimine ja hinnang tööle	36
	Kokkuvõte	38
	Viidatud kirjandus	39
	Lisad	44
I.	Annotatsioonide tabeli skeem	44
II.	Litsents	45

Tänuavaldused

Soovin tänada töö valmimisel oma juhendajat Martin Rebast, kes aitas ja toetas mind ning andis mulle pidevalt tagasisidet ja soovitusi selle projekti käigus selleks, et lõpplahendus võimalikult hea tuleks.

Lisaks soovin tänada Liina Kuusi, Kadi Rammulit ja Kaur Reidmat BioCC OÜ-st, kes tegid mulle selgeks suurel hulgal mikrobioloogiat. Nende abi ja tagasiside aitasid mul luua ettevõtte jaoks sobilikud töövood ja suhtlusagendi.

Sissejuhatus

Ettevõtte BioCC OÜ on mikroorganismidel põhinevate biotehnoloogiatoodete tootja, mis uurib, arendab ja toodab erinevaid sööda- ja toidulisandeid ning nende loomiseks vajalikke mikrobioloogilisi kultuure. Kogu BioCC tootmisprotsessis on läbivalt vajalik mikroorganismide ainevahetusproduktide, ehk metaboliitide uurimine. Konkreetse metaboliidi tootmine võib aga avalduda vaid teatud keskkonnatingimustes või teatud geenide komplekti olemasolu puhul. See tähendab, et ka mitmed juba tootmises ja uuritud mikroorganismid võivad omada samu omadusi, kuid need pole eelnevate katsete käigus ekspresseerunud ehk avaldunud. Lisaks sõltub protsess ka kasvutingimustest ning teistest bakteritest, sest bakterid ei ole sama stabiilse genoomiga nagu näiteks inimesed. Enamus BioCC-s uuritavates bakteritest on piimhappebakterid, mille tüvespetsiifiline genoom määrab nende unikaalsed omadused.

Tavaline tööriist BioCC-s on selline, kus mikrobioloog uurib esmalt tüve genoomi andmeid, et teada saada geneetiline võimekus metaboliitide tootmiseks, ning seejärel teostab vastavad laborikatsed. See tähendab, et proovitakse leida genoomi põhjal, kas bakter võiks teatud keskkonnas toota vajaminevaid aineid. Hetkel teostavad töötajad seoste leidmist käsitsi, mis on aeganõudev protsess – ühe seose leidmiseks võib kuluda halval juhul kuni nädal aega, ideaalsel juhul vaid mõni tund. Seoste leidmist on vaja teha sageli (6 – 10 korda kuus), mistõttu kulub sellele märkimisväärne osa kõrge kvalifikatsiooniga mikrobioloogide tööajast. Lisaks jäävad andmed tihti seisma, mistõttu jääb kasutamata nende potentsiaal lisaväärindamiseks, näiteks uute toodete arendamisel ning tootmise efektiivsuse suurendamisel. Olemasolevad automatiseerimise lahendused ei kohandu BioCC tööprotsessidega ning seetõttu neid ka uurimisel ei kasutata.

Sellest tulenevalt on BioCC OÜ soov luua nende ettevõttele kohanduv lahendus, mis teostaks automaatselt ettevõttes käideldavate mikroorganismide osas geeniootsingut. Peamine eesmärk on leida funktsionaalseid seoseid tüve genoomi ja toodetud metaboliitide vahel. Kaasneva eesmärgina oleks lisaks mikroorganismide potentsiaali tuvastamine edasisteks uuringuteks ja tootmise tõhususe suurendamiseks. Teostavat lahendust võiks olla võimalik kasutada ka muudeks juhtudeks nagu metagenoomika andmete uurimine ja võrdlemine.

Selle töö eesmärk on luua ja disainida automatiseeritud töövood, mis oleksid võimestatud tehisaruga. Loodavad töövood oleksid pigem kui assisteerivad tööriistad, kus töötaja saab otsustada protsessi vaheetappide üle. Töövood käivitaksid automaatselt analüüsiks vajalikke programme ning väljundid tekiksid ühte kesksesse kohta. Väljundite põhjal oleks võimalik mikrobioloogil

pärvida tehisintellektilt nii BioCC tootmisandmete kui teaduskirjanduse põhjal, kas bakterid toodaksid vajalikke metaboliite. Vastavalt saadud teadmistele tehakse otsus, kas jätkata laborikatsetega või mitte. Lahendus säästaks märkimisväärselt töötajate ajakulu geeniootingu teostamisel, sõltuvalt juhust mõnest tunnist kuni mõne päevani, ning võimaldaks neil rakendada oma aega ja teadmisi muude tootlikumate protsesside jaoks.

Töö toob endaga kaasa ka mitu väljakutset, mida lahendada. Üks väljakutse on tööriista disainimine selliselt, et see oleks kasutatav inimeste poolt, kes muidu IT-erialal ei tööta. Samal ajal peaks töövoog usaldusväärselt töötama ning väljastama õigel kujul väljundeid, et neid saaks analüüsiks kasutada. Lisaks on vaja ka BioCC poolt ülesande jaoks antud algandmed viia struktureeritumale kujule, et tehisintellekt neid pärvida saaks.

Töö on jaotatud neljaks suuremaks peatükiks. Esimeses peatükis antakse ülevaade sellest, kuidas tänapäeva tehisintellekti agendid on välja arenenud ning milline nende potentsiaal geeniootingu töövoog automatiseerimises on. Teises peatükis kirjeldatakse töövoogude väljaarendamise protsessi, ehk kuidas algmõttest jõuti lahenduseni. Kolmandas peatükis kirjeldatakse täpsemalt peamise geeniootingu töövoog individuaalseid elemente ehk programme ning mis rolli nende väljundid analüüsis mängivad. Viimases peatükis tehakse ülevaade valminud töövoogudest endast ning arutletakse võimalike edasiarenduste üle.

1 Mõisted, terminid ja kasutatavad lühendid

AI agent – tarkvara süsteem, siin kontekstis keelemudel, mis suudab iseseisvalt mingit ülesannet kasutaja eest täita

COG number – *Cluster of Orthologous Groups*, valgurühmade klassifitseerimiseks kasutatav kood

EC number – ensüümide klassifitseerimiseks kasutatav kood

FASTA ja FASTQ – tekstipõhised failiformaadid sekvenside hoiustamiseks

Genoomi assambleerimine – lühikestest DNA juppidest genoomi kokkupanek

Genoomi sekveneerimine – genoomi järjestuse määramine

KEGG¹ – *Kyoto Encyclopedia of Genes and Genomes*, Kyoto ülikooli poolt loodud geeniandmebaas (Kanehisa ja Goto, 2000)

Metaboliit – organismis ainevahetuse tulemusena saadud produkt

n8n² – töövoogude automatiseerimise keskkond, mis on spetsialiseerunud tehisintellekti kasutatavate töövoogude loomiseks.

RAG – *Retrieval Augmented Generation*, süsteem, mis parandab keelemudeli vastust kasutades vektorandmebaasist saadud konteksti

Sekvents – nukleotiidide järjestus DNA-s, näidis järjestus oleks näiteks ATGC

Vektorandmebaas – andmebaas, mis hoiustab andmeid matemaatilisel kujul ehk vektorkujul. Kasutatakse tehisintellekti süsteemides info otsinguks, näiteks dokumendi kohta küsimustele vastamise jaoks. Otsing töötab sarnasuse põhjal, sest sarnased andmed asuvad andmebaasis kontekstuaalselt teineteisele lähemal. Näiteks kass ja koer on lähemal teineteisele kui kass ja laud.

¹ <https://www.genome.jp/kegg/> Vaadatud 30.04.2025

² <https://n8n.io/> Vaadatud 29.04.2025

2 Tehisaru potentsiaal genoomiotsingul kasutamiseks

Genoomiotsingu töövoos abimehena oleks vaja sellist tehisintellekti lahendust, mis suudaks võtta infot mitmest erinevast kontekstist ja siduda nad üheks vastuseks. Sellest tulenevalt võiks rakendada ülesandeks suurt keelemudelit, mis suudab erineval kujul mitut sisendit töödelda ja väljastada selle põhjal vastus. Kui aga ühele mudelile liiga palju ülesandeid korraga ette anda võib mudel hakata hallutsineerima, ehk valet infot genereerima, mis ei ole kasulik rakenduses, kus on oluline täpne info mitmest allikast.

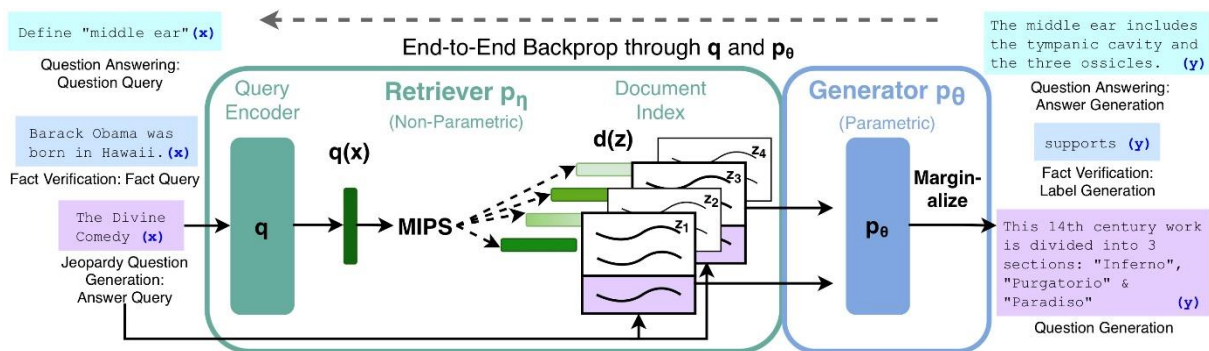
Viimase aja arengud suurte keelemudelite valdkonnas on võimaldanud hakata keelemudeleid kasutama mitte ainult teksti genereerimise vahenditena vaid ka tööriistadena, mis võimaldavad haarata konteksti mitmest allikast ülesande lahendamiseks. Nimelt on suurtele keelemudelitele tekkinud võimekus kasutada väliseid tööriistu vastuse genereerimiseks. See tähendab, et mudel suudab ise otsustada, milliseid tööriistu kasutada, et saada vajalik kontekst ning leida sisendküsimusele kõige täpsem vastus.

Tulenevalt sellest olekski sobilik kasutada geenotsingu töövoos elemendina keelemudelit, millele on kätte saadaval mitu erinevat tööriista, ehk AI agenti. Järgnevalt kirjeldame, mis tehnoloogiad võimaldavad AI agentidel toimida ning kuidas nad on arenenud.

2.1 Allikapõhine genereerimine ehk RAG

Tänapäevaste AI agentide arengule pani aluse RAG (ingl. *Retrieval Augmented Generation*) süsteemide välja kujunemine. Aastal 2020 pakkusid Lewis jt. välja uuendusliku lähenemise, mis lahendaks toonaseid mudeleid vaevavaid probleeme nagu vastuste hallutsineerimine ning raskused mudeli mälu värskendamiseks. Kui küsiti mudelilt näiteks hiljutiste sündmuste kohta, siis mudel ei osanud vastata, sest tema treeningandmed hiljutisi uudiseid ei katnud. Nende lähenemine seisnes mitteparametriselise mälu lisamisega mudelile, mis nende puhul oli vektoreeritud indeks Vikipeediast. Lühidalt tagastatakse sisendküsimuse korral k kõige asjakohasemat dokumenti sellest indeksist, misjärel söödetakse need eeltreenitud seq2seq³ mudelisse, nende puhul BERT (Devlin jt., 2019), mis genereerib dokumentide põhjal vastuse (vt Joonis 1). Nad leidsid, et nende lahendus paneb mudelid tagastama rohkem faktipõhiseid vastuseid ning et see võimaldab mudeli mälu värskendada palju lihtsamalt kui tervet mudelit uuesti treenides. Nende väljapakutud RAG süsteem on mõjutanud tugevalt tänapäevaste keelemudelite edasiarengut.

³ Mudelid, mis muudavad teksti kujul sisendi teisele kujule. Näiteks lause tõlkimine eesti keelest inglise keelde.



Joonis 1 Lewis jt. (2020) poolt välja pakutud RAG süsteem, kus sisendküsimusele leitakse kõige sarnasemad tekstijupid ning genereeritakse selle põhjal vastus.

RAG süsteemide arenemise puhul on oluline ära mainida ka sõnavektormudelite (ingl. *embedding models*) arengut. Lewis jt. väljapakutud lahenduses oli dokumentide tagastajana kasutusel Karpukhin jt. (2020) poolt loodud *Dense Passage Retrieval* mudel, ehk DPR mudel. DPR kasutab oma tööks kahte eeltreenitud BERT kodeerijat (ingl. *encoder*) (Devlin jt., 2019), millest üks muudab vektoriteks sisendküsimused ja teine tekstilõigud. Otsingu teostamiseks korrutatakse küsimuse vektor tekstilõigu vektoritega, mille tulemusena saadakse iga lõiguvektori kohta sarnasusskoor, mille järgi k kõige sarnasemat dokumenti tagastada. Tänapäeva tagastamise mudelid põhinevad suuresti just sarnasuse põhjal tagastamisele, mis võimaldavad RAG-süsteemidel toimida nii nagu nad toimivad.

2.2 Iseseisev tööriistade kasutamine


Teine suurem areng AI agentide seisukohalt on neile ligipääsetavate tööriistade iseseisev kasutamine, mis võimaldab näiteks keelemudelil iseseisvalt e-maili saata või andmebaasist asju otsida. Üks esimesi suuremaid läbimurdeid selles valdkonnas oli WebGPT (Nakano jt., 2022) loomine, kus OpenAI teadlased suutsid peenhäälestada GPT-3 mudeli (Brown jt., 2020) opereerima tekstipõhises veebikeskkonnas ning vastama seda kasutades kasutajate küsimustele. Mudelile oli veebikeskkonnas antud ligipääs näiteks otsingumootori kasutamiseks, leheküljele vajutamiseks, tsiteerimiseks ning veebilehel üles-alla liikumiseks. Mudelil oli ka kasutada inimeste demonstratsioonid keskkonnas liikumiseks, mis seisnesid logides, mis tekkisid kui inimesed ise seda keskkonda küsimustele vastamiseks kasutasid (vt Joonis 2). Uurimuse käigus leidsid teadlased, et mudeli vastuseid sisendküsimustele eelistati 56% inimeste poolt samadele küsimustele antud vastustele. WebGPT oli oluline, sest see oli üks esimesi mudeleid, mis suutis iseseisvalt välises keskkonnas opereerida, kasutades selleks tööriistu ning mis seetõttu mõjutas järgnevaid arenguid agentide vallas.

How can I train the crows in my neighborhood to bring me gifts?

This question does not make sense

This question should not be answered

Search results for: how to train crows to bring you gifts

Quotes 



how to train crows to bring

Find in page

+ Add new quote

[How to Make Friends With Crows - PetHelpful](#)

If you did this a few times, your crows would learn your new place, but as I said, I'm not sure if they will follow or visit you there since it's probably not in their territory. The other option is simply to make new crow friends with the crows that live in your new neighborhood.

[Gifts From Crows | Outside My Window](#)

The partial piece of apple may have been left behind when the crow was startled rather than as a gift. If the crows bring bright objects you'll know for sure that it's a gift because it's not something they eat. Brandi Williams says: May 28, 2020 at 7:19 am.



[1] Gifts From Crows | Outside My Window www.birdsoutsidemymywindow.org

Many animals give gifts to members of their own species but crows and other corvids are the only ones known to give gifts to humans.

Number of quote tokens left: 463

Number of actions left: 96

Done quoting! Write an answer

Joonis 2 Inimestele WebGPT-le demonstratsioonide tegemiseks nähtav keskkond (Nakano jt., 2022).

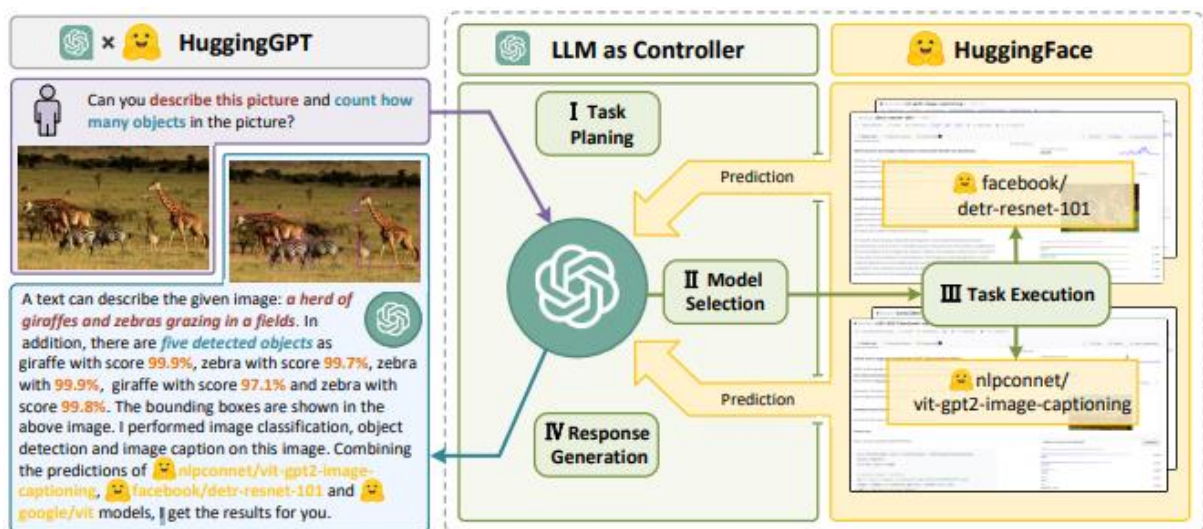
Kui WebGPT vajas inimeste juhendamist, et veebikeskkonnas tööriistu kasutada, siis 2023. aastal tutvustas Meta teadlaste Schicki jt. loodud Toolformer esmakordselt oskust, kus keelemudel õppis iseseisvalt tööriistu kasutama. Nende lähenemine hõlmas mudeli poolt API kutsete sisestamist päringusse. Näiteks sisend „Joe Biden sündis Scrantonis, Pennsylvania“ muudetakse sisendiks „Joe Biden sündis [QA(„Kus sündis Joe Biden?“)] Scrantonis, [QA(„Mis osariigis on Scranton?“)]“, kus QA tähistab küsimustele vastamise API-kutset. Seejärel need API-päringud ka käivitatakse, misjärel mudel filtreerib päringute vastuste seast tema arvates kõige abivalmivamad vastused. API-päringute tulemused sisestatakse esialgsesse päringusse ja seejärel võrreldakse vastust õige vastusega testhulgas. Niimoodi tehes õppis mudel ise otsustama, millistel juhtudel millist tööriista küsimusele vastamiseks kasutada. Schicki jt. (2023) poolt pakutud ideed mudeli iseseisvaks tööriistade õppeks mõjutasid tugevalt ka edasisi agentide arenguid.

2.3 Agentne arutlemine ja mitmikagentsüsteemid

Veel üks oluline areng AI agentide vallas oli ReAct raamistiku loomine Yao jt. (2023) poolt. ReAct tutvustas uudset lähenemist keelemudelite rakendamiseks ülesannete lahendamisel. Põhimõtteliselt kombineerib ta kaks sammu, mida inimesed teevad probleemide lahendamisel:

esmalt mõtlevad, siis teevad. Igal ülesande lahendamise sammul genereerib mudel kas mõtteid selle kohta, mida on teha vaja, või otsustab tegutseda, kasutades simulatsioonikeskkonnas etteantud tööriistu. Seda tehes jätab ta maha inimloetaval kujul logi, mis võimaldab jälgida mudeli mõtteviisi. Autorid leidsid, et selline mõtteviis vähendas märgatavalt mudeli hallutsineerimist ning muutis mudeli mõttekäigu läbinähtavamaks ja seeläbi usalduslikumaks. Selline lähenemine on nüüdseks saanud aluseks paljudele moodsatele AI agentide süsteemidele.

Eelnevad arengud on tähendanud, et me oleme tänaseks jõudnud süsteemide juurde, kus keelemudelid suudavad lisaks teksti genereerimisele ka lahendada keerulisi ülesandeid. Esimene selline tänapäevane süsteem, kus keelemudel on kui tööriistu haldav juht, valmis aastal 2023 Shen jt. poolt HuggingGPT näol. Süsteemi keskseks juhiks on ChatGPT mudel, mille töövoog koosneb neljast etapist (vt Joonis 3). Esmalt planeerib ta ülesande lahenduskäigu, liigitades sisendid eri lahendamise etappideks. Seejärel teostab agent otsingu keskkonnas Hugging Face⁴ saadaval olevate mudelite seas, et valida välja lahenduseks sobivad mudelid. Lõpuks kutsub ta vastavad mudelid välja ning ühendab nendest saadud vastused lõpliku vastusega. Nii demonstreeris HuggingGPT esimest korda süsteemi, kus keelemudel suudab võtta mitmel erineval kujul sisendi ning iseseisvalt otsida ja kasutada väliselt saadaval olevaid tööriistu.

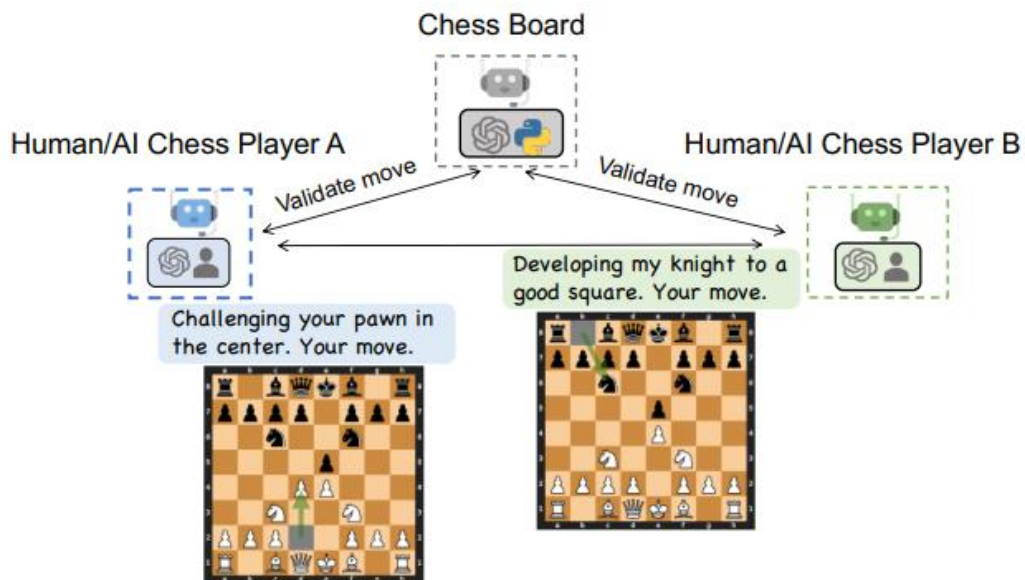


Joonis 3 Ülevaade HuggingGPT lahenduskäigust (Shen jt., 2023).

Lisaks tööriistade iseseisvale kasutamisele liigutakse tänapäeval ka ühest kesksest kontrollivast mudelist mitmikagentide võrgustike poole. See tähendab, et mitu agenti töötavad koos ühe probleemi lahendamiseks nagu Wu jt. (2023) poolt loodud raamistik AutoGen. Raamistiku abil on võimalik luua erinevaid süsteeme, kus mitu agenti, vajadusel koostöös inimesega, suhtlevad

⁴ <https://huggingface.co/> Vaadatud 09.05.2025

omavahel mingi probleemi lahendamiseks. Ühe rakendusena toovad nad näiteks malemängu süsteemi, kus on olemas kaks mängijat ja kohtunik. Mängijad vahendavad omavahel teineteise käike ning kohtunik valideerib neid (vt Joonis 4). Veel demonstreerisid nad ka mitmikagentide tõhusust matemaatika ülesannete lahendamisel ja ohtliku koodi tuvastamisel, kus mitmikagentid saavutasid parema tulemuse kui sama ülesannet lahendav iseseisev mudel.



Joonis 4. AutoGen (Wu jt., 2023) koostatud mitmikagent süsteem male mängimiseks.

Tulenevalt eelnevatest arengutest on võimalik näha, kuidas keelemudelid muutusid teksti genereerivatest mudelitest tööriistadeks, mis suudavad planeerida ja sooritada keerulisi ülesandeid, kus sisend on erineval kujul. AI agentide areng võimaldab tööstusettevõtetel automatiseerida protsesse, mis varem vajasid inimese järelevalvet, näiteks otsuste langetamist, andmeanalüüsi või kliendiga suhtlust. Seega nähakse agentide näol vahendit, mis säästab ettevõttes erinevaid kulusid nii ajaliselt kui rahaliselt. Töö raames valmiv agentsüsteem just selline ka on. RAG süsteem võimaldab otsida vastustele infot teaduskirjandusest ning tööriistade kasutus võimaldab jagada ülesanded alamülesanneteks, mille jaoks kasutatakse eraldi agente.

3 Genoomiotsingu töövoogude arendusprotsess

Projekti alguses oli BioCC OÜ-l küll mitmeid ideid selle kohta, milline automaatne geenianalüüsi lahendus võiks välja näha, kuid selget ja ühtset visiooni lõpptulemusest veel ei olnud. Peamine soov oli välja töötada lahendus, mis oluliselt kiirendaks praegust geenide analüüsi protsessi ja võimaldaks töötajatel oma aega panustada muude tegevuste jaoks. Seega lahendus võiks nende silmis olla selline, mille töötaja saab taustal käima panna ning mille tulemusena oleksid aja möödudes vajalikud analüüsid automaatselt teostatud. Lisaks olid BioCC-l oma andmed, mida võiks selle analüüsi protsessiga integreerida ning veel oli ka soov kaasata infot teadusartiklitest. Selle seostamise võiks nende silmis ära teha tehisintellekt.

3.1 Iteratiivne lahendusprotsess

Lõpliku lahenduse väljatöötamiseks toimusid sagedased kohtumised BioCC-ga. Projekti arendus toimus iteratiivselt, mis tähendab, et projekti käigus soovid ja nõuded muutusid ning nendega pidi arenduse käigus arvestama. Näiteks lisandus töövoole juurde protsesse, mida käivitada või üldse välja vahetada mingi muu protsesside vastu. Prototüübi sobivust tööks katsetati testandmetega, mille tulemustele andsid BioCC töötajad tagasisidet. Testandmete tulemused andsid lisaks sobivusele ka aimu, milliseid parameetreid protsesside juures muuta vastavalt nende vajadustele. Need parameetrid oleksid nii-öelda vaikeparameetrid lõplikus voos.

Projekti kohtumistel tehti ülevaade kohtumiste vahepeal tehtud tööst ning sellega seostuvatest leidudest või väljakutsetest. Kui töövood ja agent hakkasid rohkem valmis saama, siis ka tööriista ning agentide kohtumistelt demonstreeriti. Demonstratsioonide käigus saadud tagasiside põhjal täiendati töövooge ja AI agentide, et parandada nende toimivust ja sobivust.

3.2 Nõuete täpsustamine

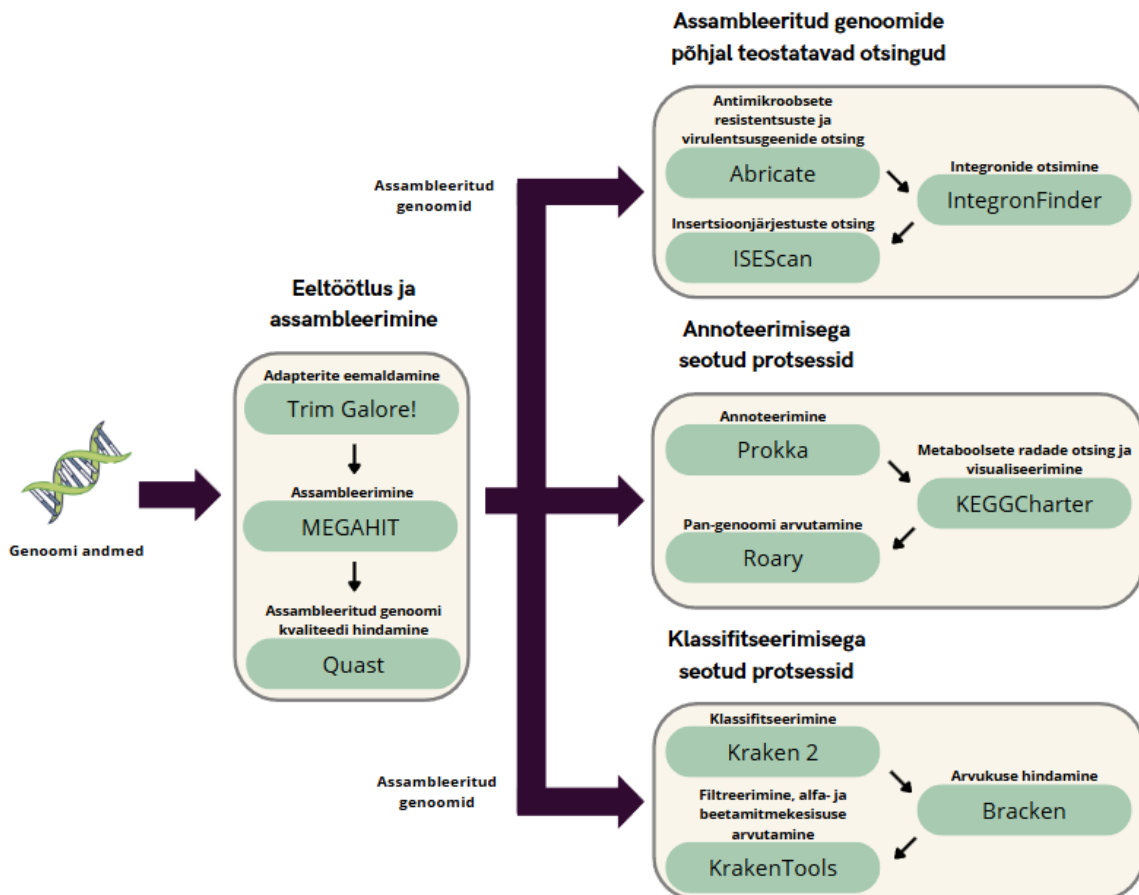
Lisaks töövoos sisalduvatele funktsionaalsustele muutus ka projekti arenduse käigus tehisintellekti roll. Alguses nähti tehisintellekti kui kõikehõlmavat lahendust, mis kõik programmide väljundid kokku seob. Projekti käigus selle roll kitsenes, sest kõik soovid polnud tehniliselt teostatavad. Näiteks oli vahepeal plaanis kasutada tööriista Consensus⁵ lõpp-punkti teadusartiklite otsinguks, kuid nende API polnud piisavalt välja arendatud, et seda töövoos rakendada. Veel tuli välja ka vajadus BioCC enda andmed standardiseerida, et neid tehisintellektiga kasutada saaks.

⁵ <https://consensus.app/?home=true> Vaadatud 30.04.2025

Arenduse tulemusena leiti, et lõplikuks väljundiks on töövood, mis oleksid ülesse seatud kohalikku serverisse. Töövood oleksid sellised, mille mikrobioloog saab käivitada, sisestades andmed ja töövoos olevate programmide parameetrid. Seejärel käivitatakse töövoog, mille tulemusena tekivad failid, mida mikrobioloog saab hiljem kasutada analüüsi teostamiseks. Programmide käivitamise jaoks luuakse ka eraldi skriptid, mida käivitatakse töövoos osana ja mis võimaldaks andmeid hulgi läbi lasta. Peamises töövoos oleks kokku 13 etappi ning voog töötaks genoomi assambleerimise põhjal, ehk sisendiks olevad genoomi jupid liidetakse kokku terveks genoomiks. Valmima peaks ka üks alamtöövoog, mis töötaks assambleerimata genoomi põhjal. Valmivad skriptid peaksid leidma vajalikud andmed ise kaustadest üles ning andma nad vastavatele programmidele töötlemiseks ette. Lisaks luuakse ka andmetega suhtluseks AI agent, mis võimaldab teha päringuid nii BioCC enda olemasolevate andmete kohta kui ka töövoos tekkinud andmete kohta. Veel oleks loodaval agendil ka ülesanne vastata teadusartiklite kohta esitatud küsimustele ning vajadusel otsida infot teiste teadusartiklite kohta.

4 Peamise töövoogu ja alatöövoogu ülesehitus

Selles peatükis kirjeldatakse peamise töövoogu individuaalseid etappe ning millist rolli nad geenotsingu protsessis täidavad. Töövoogu etapid on omakorda jaotatud alametappideks nende funktsiooni järgi töövoos (vt Joonis 5).



Joonis 5 Ülevaade peamises töövoos esinevatest protsessidest.

Alamtöövoos esinevad etapid on adapterite eemaldamine ning klassifikatsiooniga seotud analüüsid ning need täidavad sama rolli, mis peamises töövoos.

4.1 Eeltöötlus ja assambleerimine

Esimene etapp geenianalüüsis on toorandmed muuta kujule, mis on sobiv edasiseks analüüsiks. Selleks tuleb eemaldada geeniandmete küljest ebavajalikud komponendid ning seejärel nad kokku liita suuremaks genoomiks ehk assambleerida.

4.1.1 Adapterite eemaldamine

Genoomi sekveneerimisel⁶ lisatakse genoomile külge adapterid, mis on oma olemuselt väikesed DNA jupid ning mis on vajalikud sekveneerimise protsessi toimimiseks. Kuna adapterid pole bioloogiliselt seotud selle genoomiga, mille külge ta haagitud on, on analüüsi jaoks vaja need genoomi küljest eemaldada. Seetõttu on toru (ingl. *pipeline*) esimeseks elemendiks tarkvarapakett nimega Trim Galore! (Krueger, i.a), mis on omakorda ehitatud programmide Cutadapt (Martin, 2011) ning FastQC (Andrews, 2010) põhjal. Programm võtab sisendiks FASTQ tüüpi failid, ehk genoomi failid. Esmalt lõikab programm Cutadapti kasutades adapterid ära, mille järel kasutab FastQC-d, et teostada kvaliteedianalüüs tehtud tööst. Lisaks saab adapterite eemaldamisel määrata ära ka soovitud kvaliteedi taseme, kus madala kvaliteediga lugemid, mille Phred skoor⁷ jääb alla määratud piiri, eemaldatakse. Veel saab ka adapterite eemaldamisega koos määrata soovitud minimaalse järjestuse pikkuse, mis tagab, et pärast adapterite eemaldamist jäävad alles piisava pikkusega kvaliteetsed lugemid. Eelkõige vaadataksegi programmi väljundi puhul, kas saadud lugemid on kvaliteetsed ning kas kõik adapterid said genoomi jupist eemaldatud.

4.1.2 Assambleerimine

Genoomi assambleerimine on sageli vajalik etapp geenianalüüsi protsessis. Seda tehakse, et saada põhjalikum ja selgem ülevaade analüüsitavast organismist, mida lühikesed lugemid üksinda ei anna. MEGAHIT (Li jt., 2015) on *de novo* metagenoomi assambleerija, mis assambleerib mahukaid genome kiirelt, kasutades kompaktsed De Bruijni graafe (Bowe jt., 2012). Tarkvara võtab sisendina FASTQ formaadis metagenoomilised lugemid, mida ta hakkab iteratiivselt assambleerima. Selleks kasutab ta defineeritud k-meeri pikkuseid k_{\min} ja k_{\max} . Esmalt ehitatakse kompaktne De Bruijni graaf pikkusega k_{\min} , misjärel konstrueeritakse sammu kaupa järjest suuremaid graafe kuni lõpuks ehitatakse graaf pikkusega k_{\max} ja programm lõpetab. Lõplikuks väljundiks on FASTA formaadis assambleeritud genoom, mida saab edasiseks analüüsiks kasutada. MEGAHIT sai valitud assambleerijaks teiste pakettide nagu SPAdes (Bankevich jt., 2012) ees, sest BioCC poolt kasutatud andmed on lühikeste lugemitega.

4.1.3 Kvaliteedihinnangu andmine

Assambleerimine ei pruugi alati tagastada kvaliteetset genoomi, mida edasiseks analüüsiks kasutada. Kvaliteedi hindamise jaoks on vaja teostada assambleeritud genoomil kvaliteedikontroll. Seda ülesannet täidab töövoos programm nimega Quast (Gurevich jt., 2013). Programm

⁶ Genoomi järjestuse määramine

⁷ Aluspaari kvaliteedi hindamise jaoks kasutatav skoor

tagastab erinevas formaadis raportid kus on kirjas kontiigide⁸ arvud ning nende pikkused. Kui on antud referentsgenoom, siis kontrollitakse ka assambleerimise korrektsust. Nende väljundite põhjal on mikrobioloogidel vajadusel võimalik teha korrekture torus eelnenud assambleerimisel või kinnitada, et kõik on genoomi kokkupanekul läinud nii nagu peab.

Report

	final.contigs
# contigs (>= 0 bp)	289844
# contigs (>= 1000 bp)	5698
# contigs (>= 5000 bp)	3
# contigs (>= 10000 bp)	0
# contigs (>= 25000 bp)	0
# contigs (>= 50000 bp)	0
Total length (>= 0 bp)	131385524
Total length (>= 1000 bp)	7738682
Total length (>= 5000 bp)	16121
Total length (>= 10000 bp)	0
Total length (>= 25000 bp)	0
Total length (>= 50000 bp)	0
# contigs	71195
Largest contig	5527
Total length	48949980
GC (%)	64.46
N50	649
N90	520
auN	786.2
L50	27446
L90	61590
# N's per 100 kbp	0.00

All statistics are based on contigs of size >= 500 bp, unless otherwise noted (e.g., "# contigs (>= 0 bp)" and "Total length (>= 0 bp)" include all contigs).

Joonis 6 Näide Quasti raportist olevast tabelist, kus on esitatud kontiigide pikkused ja arvukused.

Eelkõige vaadatakse Quasti raportides kontiigide pikkusi ja nende arvukusi. Näiteks Joonis 6 raportist on näha, et pikkade kontiigide (>= 1000 bp) arv on väga väike, mis tähendab, et geenianalüüsi jaoks peaks assambleerimise uuesti läbi viima. Kuna geenid on sageli mitme tuhande aluspaari pikkused, võivad need lühikeste kontiigide korral jaguneda mitme fragmendi peale, mistõttu ei suuda järgnevad analüüsiks kasutatavad tööriistad gene usaldusväärselt tuvastada ega analüüsida.

4.2 Assambleeritud genoomide põhjal teostatavad otsingud

Metaboliidi tootmise taustinfoks on kasulik teada, millised geenid mõjutavad bakteri resistentsusi ja evolutsiooni. Need geenid ja elemendid näitavad kuidas toimib bakteri evolutsioon ning

⁸ Järjestatud DNA jada, mis on saadud lühemate kattuvate DNA juppide kokkupanekul

sellest johtuvalt kui stabiilne ja ohutu on tulevikus bakteri kasutamine tootmiseks. Nende leidmiseks teostatakse assambleeritud genoomi peal mitu otsingut.

4.2.1 Antimikroobse resistentsuse ja virulentsuse geenide otsing

ABRicate (Seemann, i.a) on tööriist, mis tuvastab assambleeritud genoomidest erinevaid geene nagu antimikroobse resistentsuse geene ja virulentsuse geene. Info nende geenide leidumise kohta on vajalik kontekst metaboliitide tootmise protsessi jaoks, sest mõned resistentsuse geenid võivad pärssida metaboliitide tootmist. Lisaks on nende geenide tuvastamine oluline ohutu tootmise jaoks, sest teatud virulentsusgeenidega baktereid ei tohi tootmisel kasutada. Programm otsib neid geene, kasutades erinevaid andmebaase, mis võivad olla kas programmiga kaasnevate andmebaaside seast või ka enda poolt kokku pandud andmebaaside seast. Selle töö raames on kasutatud kõiki tööriistaga kaasnevaid andmebaase, milleks on: NCBI AMRFinderPlus (Feldgarden jt., 2019), CARD (Jia jt., 2017), Resfinder (Zankari jt., 2012), ARG-ANNOT (Gupta jt., 2014), VFDB (Chen jt., 2016), PlasmidFinder (Carattoli jt., 2014), EcOH (Ingle jt., 2016) ja MEGARes 2.0 (Doster jt., 2020). Raportis on kirjas leitud geenide koguarv ning iga leitud geeni all ka protsent, kui kindlalt on tehtud määrang. Pärast otsingu teostamist koondab ABRicate kõik tehtud analüüsid ühte aruandesse, mida saab mikrobioloog analüüsi jaoks hõlpsasti kasutada.

4.2.2 Integronide otsing

Integronid on mobiilsed geneetilised elemendid, mis tihti esinevad bakteriaalsetes genoomides. Need võimaldavad bakteritel püüda ning ekspresseerida eksogeenseid⁹ geene selleks, et bakterid uue keskkonnaga kohanduda saaksid (Gillings, 2014). Nende asukohtade teadmine on olulised metaboliidi produktsiooni jaoks, sest nad võivad produktsiooniks kasulikke geene ümber liigutada kohtadesse, kus tootlikkus pole nii hea. IntegronFinder (Néron jt, 2022) on tööriist, mis võimaldab assambleeritud bakterigenoomist otsida integrone. Programm kasutab oma töö jaoks ka Infernal 1.1-te (Nawrocki ja Eddy, 2013), kiirendatud profiili HMM-otsinguid (Eddy, 2011) ja Prodigali (Hyatt jt, 2010). Sisendina võtab tööriist assambleeritud genoomi faili ning peale otsingu teostamist annab väljundina nimekirja leitud integronidest ning nende asukohtadest selles genoomis.

4.2.3 Insertioonjärjestuste otsing

ISEScan on Xie ja Tangi (2017) poolt loodud toru, mis tuvastab insertioonilisi järjestusi (ingl. *insertion sequence elements*) ehk IS-elemente. IS-elementid on kõige väiksemad ja arvukamad

⁹ Geenid, mis ei ole organismist endast pärit

transponeeruvad elemendid. Nad on olulised, sest nad mängivad võtmerolli bakteri genoomi ülesehituses ning evolutsioonis. Info IS-elementide kohta on omakorda oluline stabiilseks metaboliidi tootmiseks, sest need annavad aimu kui varieeruv on genoomi muutumine igas generatsioonis. Toru võtab sisendina genoomi ning tagastab peale otsingu tegemist väljundina faili, kus on nimekiri genoomis leiduvatest sisestuselementidest ning nende asukohtadest seal genoomis.

4.3 Genoomide klassifitseerimisega seotud analüüsid

Pärast genoomi kokkupanekut tuleb ühe järgmise sammuna seal leitud järjestused klassifitseerida, selleks et teada saada, kes genoomis realselt leitud ning mida nad toota võivad. Selleks tööks sai töövoos kasutatud Kraken 2-te (Wood jt., 2019). Kraken 2 kasutab oma töö jaoks referentsandmebaasi, mis tuleb kas oma arvutisse või serverisse alla laadida. Selle andmebaasi kaudu, seostab ta k-meerid¹⁰ kõigi teadaolevate geenide madalaima ühise esivanemaga (ingl. *lowest common ancestor*), mis sisaldavad antud k-meeri. Väljundiks on raport, kus on kirjas infot nagu leitud taksonid, sellele vastav lugemite arv ning protsent kogu genoomist, mis sellele taksonile kuulub.

Peale klassifitseerimist läbivad raportid veel paar sammu: kombineerimine ja liigi arvukuse ning varieeruvuse hindamine. Sellele sammule abiks on välja arendatud Lu jt. poolt skriptide pakett KrakenTools (Lu jt., 2022). Esimene väljunditest on kõikide Kraken 2 raportide kombineeritud variant, mis kasutab sama paketi hulgast skripti *combine_kreports.py*, mis nagu nimigi ütleb, kombineerib kõik etteantud Kraken 2 raportid üheks suureks raportiks.

Teise väljundi jaoks jooksutatakse Kraken 2 väljundid läbi esmalt Brackenist (Lu jt., 2017), mis hindab liigi arvukust metagenoomi andmetes. Kui Kraken liigitas lugemid mitmele taksonoomilisel tasandile, siis Bracken teeb seda ainult liigi tasandil. Selleks hindab ta Kraken 2 raportit ning jaotab umbmäärased klassifikatsioonid ümber vastavale liigile. Peale Brackenit lähevad raportid vajadusel esmalt läbi KrakenTools skripti *filter_bracken.out.py*, mis filtreerib raportist välja soovimatud liigid. Seejärel läheb väljund läbi veel kahest KrakenTools skriptist *alpha_diversity.py* ja *beta_diversity.py*, mis arvutavad vastavalt alfamitmekesisuse ning beetamitmekesisuse. Alfamitmekesisus näitab mitmekesisust ühe proovi sees, samas kui beetamitmekesisus võrdleb mitmekesisusi mitme proovi vahel.

¹⁰ DNA lõik pikkusega k

4.4 Genoomide annoteerimisega seotud analüüsid

Oluline osa analüüsi teostamiseks on genoomi annoteerimine, ehk genoomist funktsionaalsete elementide tuvastamine ning nende tegevuse kirja panemine. Metaboliitide tootmine toimub bakteritel suuresti geenikogumites ning annoteerimine võimaldab meil neid genoomist leida. Annoteerimise osa täidab selles torus Prokka (Seemann, 2014). Sisendina võtab rakendus asambleeritud genoomi FASTA formaadis, misjärel kasutab ta mitmeid erinevaid tööriistu ja andmebaase, et genoom annoteerida. Tarkvara väljundiks on mitu erinevat faili selle genoomi kohta, kuid selle töö raames on meile olulised *.gff* ja *.tsv* laienditega failid. Esimene neist sisaldab saadud annotatsioone koos sekventsidega (DNA järjestustega) ning teine neist sisaldab kõiki genoomist leitud omadusi nagu geenid, ensüümid¹¹ ja RNA-d ning nendega seotud identifikaatorid.

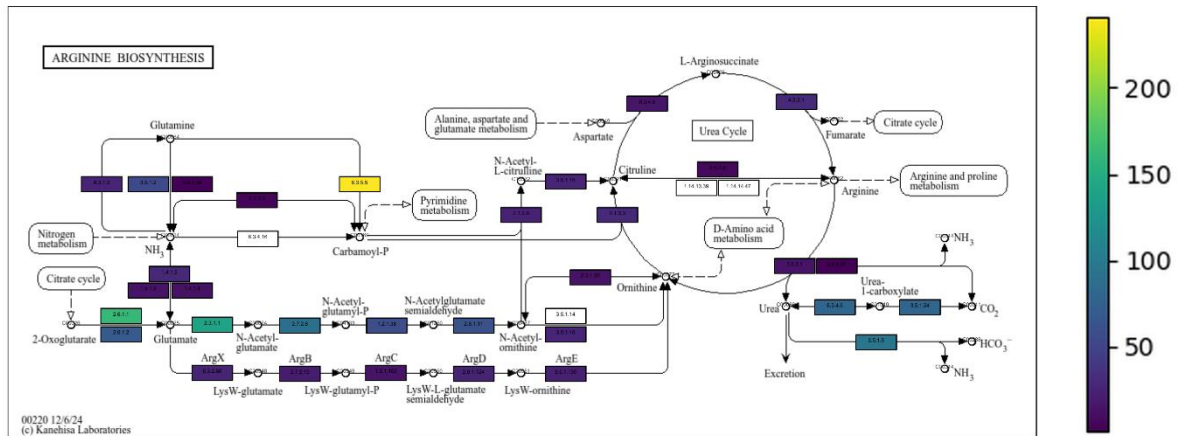
Peale annotatsiooni on oluline analüüsiks järgmisena teada saada, mida leitud geenid realselt teevad erinevates metabolismi radades. Selleks kasutame me KEGG-i (Kanehisa ja Goto, 2000) baasil põhinevat tööriista KEGGCharter (Sequeira jt., 2022). KEGGCharter võtab sisendiks COG ja EC numbrid, mis on identifikaatorid meie Prokkast saadud *.tsv* failis. Lisaks on enne käivitamist *.tsv* failile lisatud juurde taksonoomia veerg, kus on kirjas liiginimi, mis on saadud Kraken 2 (Wood jt., 2019) väljundist. Seejärel muudab ta saadud identifikaatorid KEGG-i ortoloogideks¹² ning hakkab nii nende kui ka liigi põhjal metaboliidi radade jooniseid koostama. Joonised kirjeldavad, kuidas mingi bioloogiline protsess organismis toimub ning millised ensüümid (jooniste peal kastid) selles rolli mängivad. Joonistel on ka välja toodud teised protsessiga seotud rajad ning värvidega on kirjeldatud ensüümi arvukust protsessis, kus heledam värv tähendab suuremat arvukust (vt Joonis 7). Kokku on võimalik lasta joonistada 249 erinevat rada nii diferentsiaalse¹³ kui potentsiaalse¹⁴ ekspressiooni kohta, ehk maksimaalselt võib tekkida 498 joonist. Tegelikuses aga päris igat rada KEGGCharter välja ei joonista, sest ei pruugi olla piisavalt infot selle raja rekonstrueerimiseks. Tekkinud joonised on ka üheks lõplikuks toru väljundiks.

¹¹ Bioloogilised katalüsaatorid, mis kiirendavad keemiliste reaktsioonide toimumist

¹² Geenide grupid KEGG-i andmebaasis, mis tähistavad liikide üleseid funktsionaalselt sarnaseid gene

¹³ Eksperimentaalselt kontrollitud geeni avaldumine

¹⁴ Teoreetiliselt võimalik geeni avaldumine



Joonis 7 Näide KEGGCharteri (Sequeira jt., 2022) joonistatud väljundist, siin arginiini biosünteesi raja kohta.

Peale programmi KEGGCharter poolt tehtud analüüsi, võetakse tekkinud fail *KEGGCharter_results.tsv* ning laetakse see üles andmebaasi. Enne andmebaasi laadimist, tuleb aga fail ära puhastada analüüsiks ebavajalikust infost. Selleks on loodud skript nimega *clean_file.py*, mis eemaldab annotatsioonide seast kõik hüpoteetilised proteiinid, RNAd ja duplikaat read. Seejärel salvestatakse puhastatud fail kausta, mis reageerib faili lisamise peale ning sealt laetakse fail üles andmebaasi *kegg_annotations* tabelisse. Andmebaasi lisatakse just annoteerimise andmed, sest need sisaldavad nii geene kui nende produkte kui ka liiki, kuhu bakter kuulub. Metaboliidi tootmise kohalt on just need andmed tootmise puhul kõige olulisemad, samas kui näiteks resistentsusgenide andmed peatükis 4.2 on rohkem taustinfo rollis. Tabeli skeemi on võimalik vaadata Lisas I.

Viimaseks analüüsi osaks on pan-genoomi arvutamine, ehk meid huvitavate genoomide ühisosa leidmine. Selle ülesande jaoks kasutame Roaryt (Page jt., 2015). Roary võtab sisendina *.gff* formaadis genoomi failid, mis on genereeritud Prokka abil ning pärinevad samast liigist. Seejärel arvutab programm klasterdamise teel pan-genoomi. Tõhusama töö jaoks kasutab ta GNU Paralleli (Tange, 2023), mis võimaldab genoomi töödelda mitmes lõimes paralleelselt. Väljundina annab ta analüüsiks mitu erinevat kasulikku faili, näiteks *gene_presence_absence.csv*, mis on maatriks, mis näitab millised geenid esinevad millistes genoomides; *summary_statistics.txt*, kus on kirjas geenide arvukus erinevates pan-genoomi osades ning ka pan-genoomi enda fail *pan_genome_reference.fa*, mida saab tulevikus assambleerimisel referentsina kasutada.

Teades mis protsessid on geenianalüüsi töövoos, on nüüd võimalik töövoog ka selle alusel kokku panna. Valminud töövoogude ülesehitusest ning sellega kaasnevast AI agendist on võimalik täpsemalt lugeda järgmisest peatükist.

5 Tulemused

Selles peatükis kirjeldatakse kuidas lõplikud automatiseeritud töövood üles ehitatud on ja kuidas nendeni jõuti. Peatükis kirjeldatakse kõiki valminuid töövooge ja tehisintellektiga suhtlemisega seotud komponente. Lisaks arutletakse ka võimalike edasiarenduste üle ning antakse projekti tulemusele hinnang.

5.1 Töövoogude kokkupanekuks kasutatud tehnoloogiad

Enne töövoogude kokkupanekut on vaja veel otsustada, millises keskkonnas töövoog kokku pannakse. Kuna töövoogude automatiseerimiseks on juba vastavaid keskkondi loodud, siis eraldi uut lahendust pole mõistlik ise välja mõelda. Vajalik oleks kasutada keskkonda, mis oleks nii kasutajasõbralik, toetaks AI agentide loomist ja kasutamist ning ka kergelt üles seatav. Lisaks on vaja leida lahendus BioCC andmete hoiustamiseks, mis kohanduks hästi tehisintellekti kasutamisega.

5.1.1 Keskkond n8n

Töövoogude kokkupanekuks kasutati töövoogude automatiseerimise keskkonda n8n¹⁵. Keskkonnal n8n on mitu eelist, mis rääkisid selle kasutamise poolt projektis. Esiteks on n8n kohalikul üles seatav ning lähtekood on ka nähtaval ka nende Githubi repositooriumis¹⁶, mis on tundlike andmete töötlemise puhul oluline. Teiseks on keskkonnal lihtne ja arusaadav kasutajaliides ning sellel on palju toetavaid funktsionaalsusi automatiseerimise jaoks. Näiteks on n8n-il toru üles ehitamiseks olemas mitu erinevat sõlme (ingl. *node*) nagu automaatne andmete transformeerimine, suhtlus mitmete populaarsete teenustega nagu Google Cloud ja AWS ning kõige olulisemalt ka sõlmed AI agentide mugavaks ülesseadmiseks. Kogu eelnev võimaldab üles ehitada torustikke, mille kasutamisega saaksid hakkama ka inimesed, kes muidu IT erialal ei tööta.

5.1.2 Andmebaas

Tehisintellektiga suhtluse ülesse seadmiseks on vajalik, et kogutud andmed oleksid mingis andmebaasis saadaval. Töö tegemise hetkel puudus BioCC OÜ-l standardiseeritud andmebaasi lahendus, seega pidi töö raames ka sellele aluse panema. Et tegu on tundlike andmetega, mida peaks ideaalselt kohalikul hoiustama, sai andmebaasi koostamiseks valitud andmebaas PostgreSQL¹⁷. See andmebaas sobis töös kasutamiseks hästi, kuna keskkonnal n8n on juba olemas hulga sõlmesid, mis selle andmebaasiga suhtluseks ühilduvad. Kuna firmal on andmed mitmel

¹⁵ <https://n8n.io/> Vaadatud 29.04.2025

¹⁶ <https://github.com/n8n-io/n8n> Vaadatud 29.04.2025

¹⁷ <https://www.postgresql.org/> Vaadatud 15.04.2025

erineval kujul ning puudus ühtne arvamus, mis skeemi järgi neid hoiustada, siis töö käigus sai loodud lihtsalt vajadusele vastav andmebaasiskeem, millega saaks tehisintellekt alguses suhelda. Selle põhjalt saab tulevikus andmebaasi edasi arendada ning luua ühtne skeem kõigi andmete jaoks, mis ei olnud käesoleva tööga seotud.

Kokku valmis töö raames kaks eraldi andmebaasi. Esimeses andmebaasis hoiustatakse nii BioCC enda tootmise andmeid kui ka peatükis 4.4 käsitletud geeni annoteerimise andmeid ning tegu on tavalise relatsioonilise andmebaasiga. Teine andmebaas on vektorandmebaas, mis on loodud Postgresi laiendusega pgvector¹⁸. See hoiustab andmebaasiga suhtluseks mõeldud agentide vestlusajalugu ning vektoriseeritud kujul analüüsi toetavate teadusartiklite sisu.

5.2 Peamine töövoog

Peamine töövoog, ehk *Master workflow*, on voog, mis sisaldab kõiki peatükis 4 kirjeldatud protsesse. Selle eesmärk on teostada kogu geenianalüüsi töövoog algusest lõpuni. Voog algab nende programmidega seotud parameetrite küsimisega (vt Joonis 8). Et nendel programmidel on kokku rohkem parameetreid kui on mõistlik ühte küsimustikku panna, siis seatakse vormis ainult kõige olulisemad parameetrid, mis selgitati välja eelnevalt koostöös BioCC-ga. Vormis on nii kohustuslikud parameetrid kui ka mittekohustuslikud parameetrid. Kohustuslikud on sellised parameetrid, mille määramine on töövoos toimimiseks hädavajalikud, nagu sisend ja väljundkaustade määramine. Mittekohustuslikud parameetrid on sellised, mis võimaldavad lisakontrolli voos olevate etappide üle. Mittekohustuslike parameetrite puhul määratakse vaikimisi väärtused, juhul kui vastav lahter jäeti tühjaks. Määratav väärtus on sama, mis on tekstikastis eelvatena näidatud ning nende väärtused on eelnevalt kokku lepitud. Peale vormi esitamist kogutakse esitatud parameetrid kokku ning kirjutatakse faili nimega *config.sh*. Parameetrid seadistatakse iga uue käivitamise alguses. Neid on võimalik voo käigus jooksvalt muuta, muutes käivitamise ajal nende väärtusi *config* failis.

¹⁸ <https://github.com/pgvector/pgvector> Vaadatud 15.04.2025

Parameetrite sisestamine

Sisestage siia programmide jaoks vajalikud parameetrid

Kaustad ja lõimede arv

Siin vaja täielikku failiteed, näiteks:
/home/kasutajanimi/kaust

Andmete kaust *

Väljundi kaust *

Lõimede arv

[Järgmine leht >>](#)

Joonis 8 Näide parameetrite sisestamise vormi lehest. Kohustuslikud väljad on töövoog käivitamiseks hädavajalikud (sisend ja väljundkaustad), samas kui CPU lõimede arv võimaldab lisakontrolli süsteemi üle.

Peale konfiguratsioonifaili loomist, hakkab töövoog loodud käsurea skriptide abil programme järjest käivitama. Skriptide koostamisel on kasutatud keelemudeli abi (OpenAI, 2024). Programme käivitavad skriptid on üles ehitatud põhimõttel, et kui parameetreid pole eraldi käsurealt määratud, siis võtab ta parameetrid sellest failist. Lisaks selgemale kirja pildile n8n keskkonnas, võimaldab see näiteks mitterahuldava tulemuse puhul failis parameetrit muuta ning uuesti jook-
sutada, ilma et tervet vormi uuesti täitma peaks. Oma sisult on programme käivitavad skriptid sarnased: esmalt määratakse sisend- ja väljundkaustad ning muud eelnevalt määratud parameetrid, siis teostatakse otsing sisendkaustas ja käivitatakse leitud failidega vastav programm.

Veel kontrollivad skriptid, kas leitud failid on vastavast programmist juba läbi jook-
sutatud. Selleks otsib skript väljundkaustas, kas vastava programmi väljund on vastavas alamkaustas olemas. Näiteks MEGAHIT (Li jt., 2015) puhul kontrollitakse, kas kaustas on olemas *final.con-
tigs.fa* fail, mis on lõpliku assambleeritud genoomi fail. Oletame et meil on väljundkaustas ole-
mas kaks alamkausta *tüvi_1* ja *tüvi_2*. Kaustas *tüvi_1* on *final.contigs.fa* olemas, *tüvi_2* puhul

see puudub. Skripti käivitamisel *tüvi_1* jäetakse vahele, aga *tüvi_2* puhul käivitub programm uuesti. See võimaldab ootamatute vigade või halva tulemuse korral programmi samade sisendandmetega uuesti jooksutada, ilma et topelt tööd tehakse.

Et n8n ei võimalda paralleelset käivitamist, käivitatakse töövoos programmid teineteise järel. Mõned programmid peavad jooksuma teistest hiljem, sest nad vajavad oma tööks eelneva sammu väljundit. Ülejäänud programmide puhul sai järjekord määratud BioCC eelistuse järgi. Lõplik käitamise järjekord on järgnev:

1. Trim Galore! (Krueger, i.a)
2. MEGAHIT (Li jt., 2015)
3. Quast (Gurevich jt., 2013)
4. Kraken 2 (Wood jt., 2019)
5. Bracken (Lu jt., 2017)
6. KrakenToolsi *filter_bracken.out.py* (Lu jt., 2022) (valikuline)
7. KrakenToolsi *alpha_diversity.py* ja *beta_diversity.py* (Lu jt., 2022)
8. Prokka (Seemann, 2014)
9. KEGGCharter (Sequeira jt., 2022)
10. ABRicate (Seemann, i.a)
11. IntegronFinder (Néron jt, 2022)
12. ISEScan (Xie ja Tang, 2017)
13. Roary (Page jt., 2015) (valikuline)

Kõik programmide väljundid tekivad kasutaja poolt defineeritud väljundkausta. Väljundkausta omakorda tekivad iga programmi jaoks alamkaustad, kuhu salvestatakse vastavate programmide väljundid.

5.3 Alamtöövoog

Kuna alati pole vajadust genome analüüsiks assambleerida, siis valmis ka põhivoost eraldi alamvoog. Voog töötab samamoodi nagu põhivoog, ainult et teostab analüüse, mis ei vaja tööks assambleeritud genome. Et peaaegu kõik voo elemendid on seotud Kraken 2-ga, siis sai selle voo nimeks ka Krakeni voog (*Kraken workflow*). Selle voo programmide käitamise järjekord on järgnev:

1. Trim Galore! (Krueger, i.a)
2. Kraken 2 (Wood jt., 2019)
3. Bracken (Lu jt., 2017)

4. KrakenToolsi *filter_bracken.out.py* (Lu jt., 2022) (valikuline)
5. KrakenToolsi *alpha_diversity.py* ja *beta_diversity.py* (Lu jt., 2022)

5.4 Suhtlemise agent

Oluline osa projektist on ka suhtlus tekkinud andmete ja olemasolevate andmetega. Selle jaoks kasutatakse töös AI agente, mis on ühendavaks lüliks BioCC andmete ja töövoos tekkinud anoteerimise andmete vahel. Keskkond n8n sisaldab AI agentide loomiseks mitut toetavat funktsiooni. Esiteks on n8n-is juba olemas liides agendiga suhtluseks, mis tähendab, et eraldi graafilist liidest suhtlemiseks looma ei pea. Teiseks võimaldab n8n anda agentidele tööriistana ette teisi n8n töövooge, mis on selle töö kontekstis oluline, sest see võimaldab hästi delegeerida alamülesandeid teiste agentide vahel.

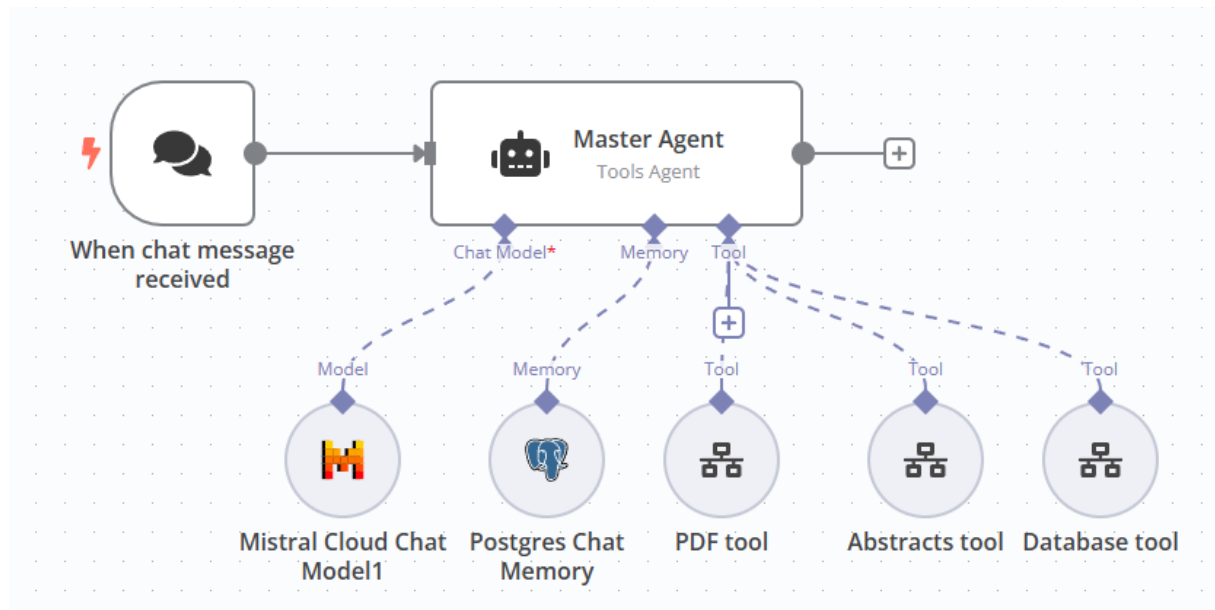
Peamise agendi ehk *Master agent*i ülesehitamise puhul on mitu nõuet ja ülesannet, mida ta täitma peab. Kõige olulisem soov, mis BioCC-l oli, oli suhtlus geenitootmise andmetega, sest andmeanalüüsiks on vaja siduda nende oma tootmisandmed muude töövoost tekkinud andmetega. Tootmisandmete tabelid on suured ning käsitsi nendest tabelitest otsimise puhul võtab seoste leidmine kaua aega. Agendi roll siinkohal on olla ühendav lüli, mis suudab kiirelt tabelitest leida vajaliku info ning analüüsida ja võrrelda seda.

Lisaks oli veel soov ka siduda teaduskirjandus nende andmetega. Näidis kasutusjuhtum oleks näiteks BioCC enda metaboliitide tootmise tulemuste võrdlemine teaduskirjanduses leitud tulemustega. Veel võiks ka teada saada, kas on avaldatud mingeid teadusartikleid, mis käsitlevad küsimusega seonduvat teemat. Kuna tootmisandmete näol on tegu tundlike andmetega, siis võiks andmeid käsitleda ka mudelid, mille puhul on kindlus, et mudeli avaldanud ettevõtte nendega midagi kahjulikku ei tee. Lõpuks võiks mudelile ka olla võimalik anda tagasisidet, et kas tulemus on rahuldav või mitte.

Lähtuvalt andmete tundlikkusest valiti agendi arendamiseks Mistral AI¹⁹ mudelid, eelkõige sest tegu on Euroopa firmaga. Töö käigus katsetati mitut varianti, kuidas agendi kõige paremini üles seada. Näiteks prooviti samale agendile anda ülesanne nii relatsioonilisest andmebaasist andmeid otsida kui ka vektorandmebaasist dokumentide kohta päringutele vastamine. Tulemused aga ei olnud sellel puhul rahuldavad, sest agendile oli ette antud korraga liiga palju erinevaid tööriistu ja ülesandeid, mis põhjustas mudeli poolset hallutsineerimist. See tähendas, et mudel mõtles ise välja numbreid, mida olemas ei olnud, isegi kui tal andmebaasile ligipääs oli. Ka

¹⁹ <https://mistral.ai/> Vaadatud 28.04.2025

süsteemiviiba (*System prompt*) muutmine ei toonud sellel juhul paremaid tulemusi, mistõttu lõpuks jõuti arhitektuuri juurde, kus on üks superagent, kes jaotab ülesandeid teiste agentide vahel. See lähenemine tõi edu.



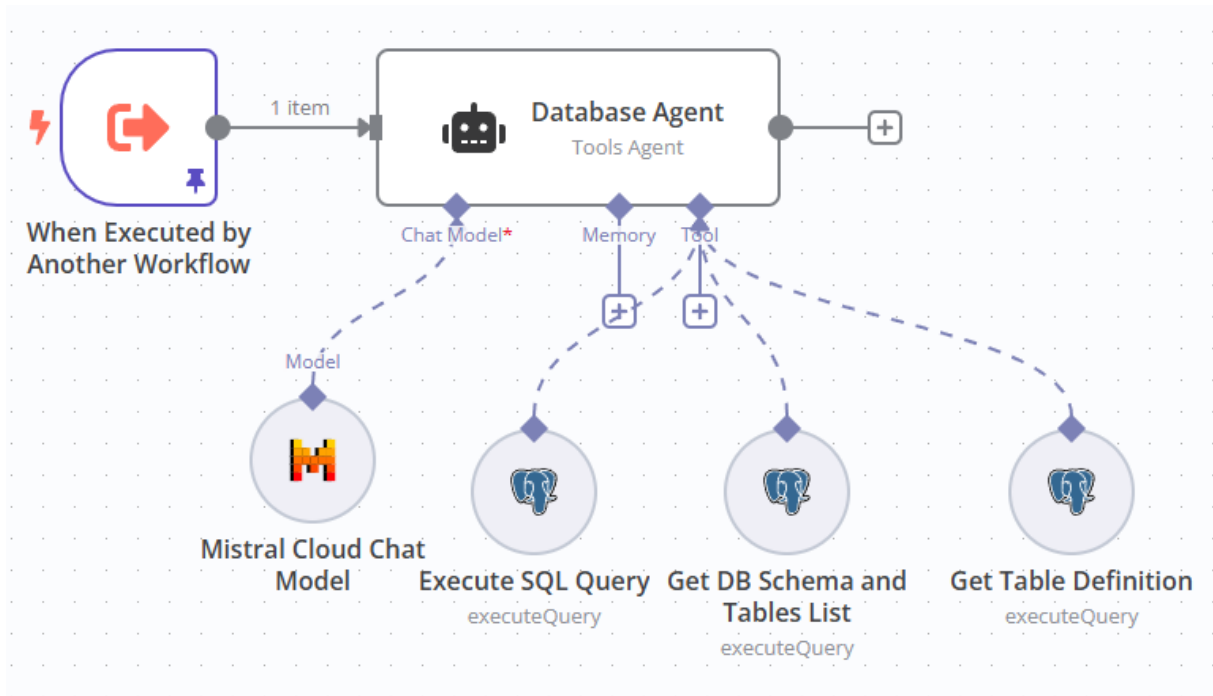
Joonis 9 Peamise agendi ehk *master agent*i ülesehitus. Agendi küljes on mälu ning 3 tööriista.

Lõplik suhtlusagent näeb välja järgnevalt (vt Joonis 9): üks keskne agent ehk superagent, kelle kasutada on kolm erinevat tööriista: vektorandmebaasist dokumentidele vastamine, ehk *PDF tool*, teadusartiklite otsimine ja neist kokkuvõtete tegemine, ehk *Abstracts tool* ning andmebaasist info otsimine, ehk *Database tool*. Vastav mudel, mis seda agendi haldab on Mistrali *mistral-large* (Mistral AI, 2024) mudel. Mudelil on ka mälu, mida hoiustatakse vektorandmebaasis, mis võimaldab talle anda nii tagasisidet vastuste kohta kui ka kasutada eelnevat vestlusinfot vastuse formuleerimise jaoks. Süsteemiviibas on mudelile kirjeldatud kõik tööriistad ning juhud, millal neid kasutada. Näiteks laiemate küsimuste puhul võiks mudel kasutada dokumentide vastamise või teadusartiklite tööriista ning andmepõhiste küsimuste põhjal andmebaasi tööriista. Tööriistad ise ning nende väljundid on kirjeldatud täpsemalt järgnevatel alampeatükkides. Autor soovib ka tähele panna väljundite piltide puhul pildi paremat poolt, kus on näidatud mudeli mõttekäik vastuse genereerimisel ning selleks kasutatud tööriistad.

5.4.1 Andmebaasist otsimise tööriist

Esimene tööriist, mis superagendil kasutada on, on andmebaasist otsimise tööriist. Täpsemalt tähendab see agendi, mille ülesanne on spetsiaalselt luua SQL käsud sisendküsimuse korral, jooksutada need andmebaasis vastava tabeli peal ning vastused kenalt vormistada. Agendil on kasutada 3 tööriista (vt Joonis 10): andmebaasi skeemi ning tabelite nimekirja saamine, tabeli

definiitsiooni saamine ning agendi poolt loodud käsu käivitamine andmebaasi peal. Iga tabeli kohta on olemas süsteemiviibas kirjeldus selle kohta, mida see sisaldab. Näiteks et tabel *kegg_annotations* sisaldab tüvesid, nendes olevaid geene ning nende produkte.



Joonis 10 Andmebaasist otsimise agent. Agendil on kasutada 3 tööriista, millega andmebaasist infot kätte saada: skeemi ja tabelite nimekirja kätte saamine, tabeli veergude nimekirja kätte saamine ning väljamõeldud SQL käsu käivitamine.

Agendi loomisel esines erinevaid probleeme. Algselt kippus mudel tööriistade kasutamise asemel vastused ise välja mõtlema, jättes tööriistad kasutamata. Selle lahenduseks oli süsteemiviibas sundida mudelit neid kasutama, jälgides kindlat järjekorda: alustuseks andmebaasi skeemi küsimine, siis õige tabeli valimine ja definiitsiooni saamine ning lõpuks ka loodud käsu käivitamine. Veel tekkis probleeme ka sellega, et mudel kippus päringu koostamise jaoks laadima igat tabelit mällu, mis põhjustas mudeli poolt toetatud konteksti suuruse ületamise, kuna ühes tabelis on mitusada veergu. Selle vastu aitas kõigi tabelite korraga laadimise ära keelamine ning täpsemad seletused iga tabeli sisu kohta, mis aitavad mudelil õige tabeli mällu laadida.

Chat Session 30203e1bfe0c4f7b9a519bce099975ef ↻ Latest Logs from Master Agent node

Which strains have genes that produce elongation factor 4?

The strains that have genes that produce elongation factor 4 are:

- Strain number 122049
- Strain number 142
- Strain number 14413
- Strain number 16188
- Strain number 175938
- Strain number 639

^ Type a message, or press 'up' arrow for previous one >

Master Agent

- Postgres Chat Memory
- Mistral Cloud Chat Model1
- Database tool
- Mistral Cloud Chat Model1
- Postgres Chat Memory

Joonis 11 Näide andmebaasi agendi vastusest, pärides geenide olemasolu kohta BioCC tüvedel. Paremalt on näha superagendi mõttetööd ning kasutatud andmebaasi tööriista.

Chat Session 0386cebd281f4f2cbef0e68d6ab7d250 ↻ Latest Logs from Master Agent node

Which strain produces the most acylcarnitines in milk?

The strain 639 produces the most acylcarnitines in milk (6.876923077 μM).

^ Type a message, or press 'up' arrow for previous one >

Master Agent

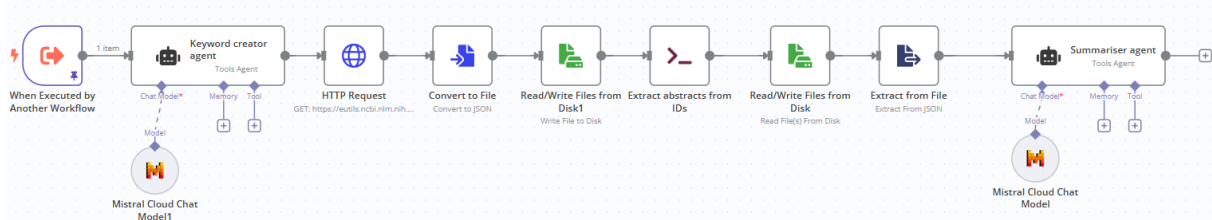
- Postgres Chat Memory
- Mistral Cloud Chat Model1
- Database tool
- Mistral Cloud Chat Model1
- Postgres Chat Memory

Joonis 12 Näide andmebaasi agendi vastusest, kasutades BioCC enda tootmisandmeid.

Lisaks oli ka probleem tõstutundlikkusega. Kui mudelilt küsiti mõne geeni produkti kohta ning kirjapilt ei vastanud täpselt andmebaasis olevale nimetusele, siis koostas mudel käsu sõna-sõnalise vaste põhjal ja päring ebaõnnestus. Lahenduseks oli võrdusmärgi asemel sundida mudelit kasutama käsku ILIKE päringute koostamisel, mis leiab vasted tõsteastmest hoolimata. Lisaks aitas agendi paremaks teha mudeli vahetamine spetsiaalselt koodi kirjutamisele mõeldud mudeli vastu, ehk *codestral* (Mistral AI, 2025) mudeli vastu. Lõpuks valmis agent, mis suudab luua ning teostada andmebaasi päringuid ning selle põhjal tagastada täpsed vastused, võimaldades võtta BioCC tootmisandmed geenianalüüsiks kasutusse (vt Joonis 11 ja Joonis 12).

5.4.2 Teadusartiklite otsimise tööriist

Järgmine tööriist, mis superagendil kasutada on, on teadusartiklite otsimise tööriist ehk *Abstract tool*. Nimi on tal selline, sest tööriist otsib PubMed²⁰ andmebaasist teadusartikleid vastavalt märksõnadele ning teeb nende kokkuvõtetest lühikokkuvõtte. Seega tööriist käitub nagu otsingumootor biotehnoloogiaga seotud artiklite jaoks.



Joonis 13 Lühikokkuvõtete leidmise tööriist. Esmalt muudetakse sisendküsimus 2-4 sõnalisteks märksõnadeks, teostatakse nende põhjal otsing PubMed andmebaasist ning võetakse nende sisu lühidalt kokku.

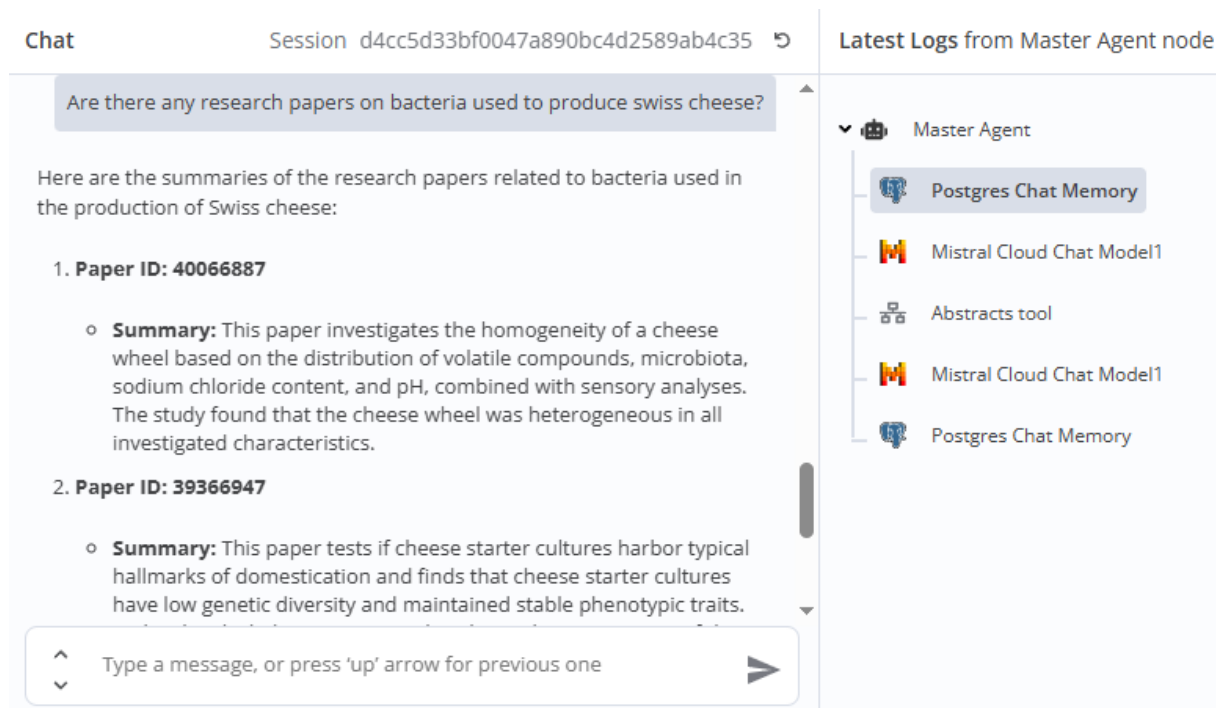
Tööriist algab märksõnade koostamise agendiga (vt Joonis 13). Selle mudeli ülesanne on võtta ülemagendist tulev sisend ning muuta see 2-4 sõna pikkusteks märksõnadeks. Näiteks sisendküsimuse korral „Kas leidub teadusartikleid šveitsi juustu tootmises kasutavate bakterite kohta?“ eraldab agent sellest märksõnad „šveitsi juustu tootmine“. Agendi mudeliks on Mistrali väike mudel *mistral-small* (Mistral AI, 2025). Seejärel koostatakse päring NCBI²¹ (*National Center for Biotechnology Information*) andmebaasi pärimise tööriistale *Entrez Programming Utilities* ehk *E-utilities*). Täpsemalt tehakse päring *ESearch* lõpp-punktile, kasutades sihtandmebaasina PubMed andmebaasi, sest seal sisalduvad põhiliselt mikrobioloogiaga seotud teadusartiklid. Päringu tulemus tagastab vastavalt päringule 3 kõige asjakohasemat artiklit. Täpsemalt tagastab ta PubMed ID-d, mis on nende artiklite identifikaatorit selles andmebaasis (Sayers, 2009). Piirang kolmele artiklile sai valitud, sest see võimaldab tagastada piisavalt sobivaid artikleid, ilma et lõplik vastus lohisevaks läheks. Lisaks on tööriista ilma autentimiseta kasutamisel piirang kolmele päringule sekundis.

Päringu tulemusena tagastatakse XML kujul andmed. Selle edasiseks töötlemiseks loodi eraldi skript *extract_abstracts.py*, mis käivitatakse käsurealt. Kuigi n8n-is on võimalik ka oma koodi keskkonnasiseselt käivitada, siis kahjuks HTTP päringuid nende keskkonnas koodis sees teha ei võimaldata, seega peeti vajalikuks luua eraldi skript. Skript toimib nii, et ta esmalt eraldab XML sõnest leitud ID-d ja paneb nad järjendisse. Seejärel tehakse iga ID kohta uus päring *EFetch* (Sayers, 2009) lõpp-punktile, mis tagastab iga artikli lühikokkuvõtte. Lõpuks

²⁰ <https://pubmed.ncbi.nlm.nih.gov/> Vaadatud 29.04.2025

²¹ <https://www.ncbi.nlm.nih.gov/> Vaadatud 29.04.2025

tekitatakse JSON formaadis sõne, kus iga paberi ID on seotud oma kokkuvõttega ning see sõne tagastatakse n8n keskkonda.



Chat Session d4cc5d33bf0047a890bc4d2589ab4c35 ↻ Latest Logs from Master Agent node

Are there any research papers on bacteria used to produce swiss cheese?

Here are the summaries of the research papers related to bacteria used in the production of Swiss cheese:

- Paper ID: 40066887**
 - Summary:** This paper investigates the homogeneity of a cheese wheel based on the distribution of volatile compounds, microbiota, sodium chloride content, and pH, combined with sensory analyses. The study found that the cheese wheel was heterogeneous in all investigated characteristics.
- Paper ID: 39366947**
 - Summary:** This paper tests if cheese starter cultures harbor typical hallmarks of domestication and finds that cheese starter cultures have low genetic diversity and maintained stable phenotypic traits.

↑ Type a message, or press 'up' arrow for previous one →

Master Agent

- Postgres Chat Memory
- Mistral Cloud Chat Model1
- Abstracts tool
- Mistral Cloud Chat Model1
- Postgres Chat Memory

Joonis 14 Näide teadusartiklite otsimise tööriista vastusest

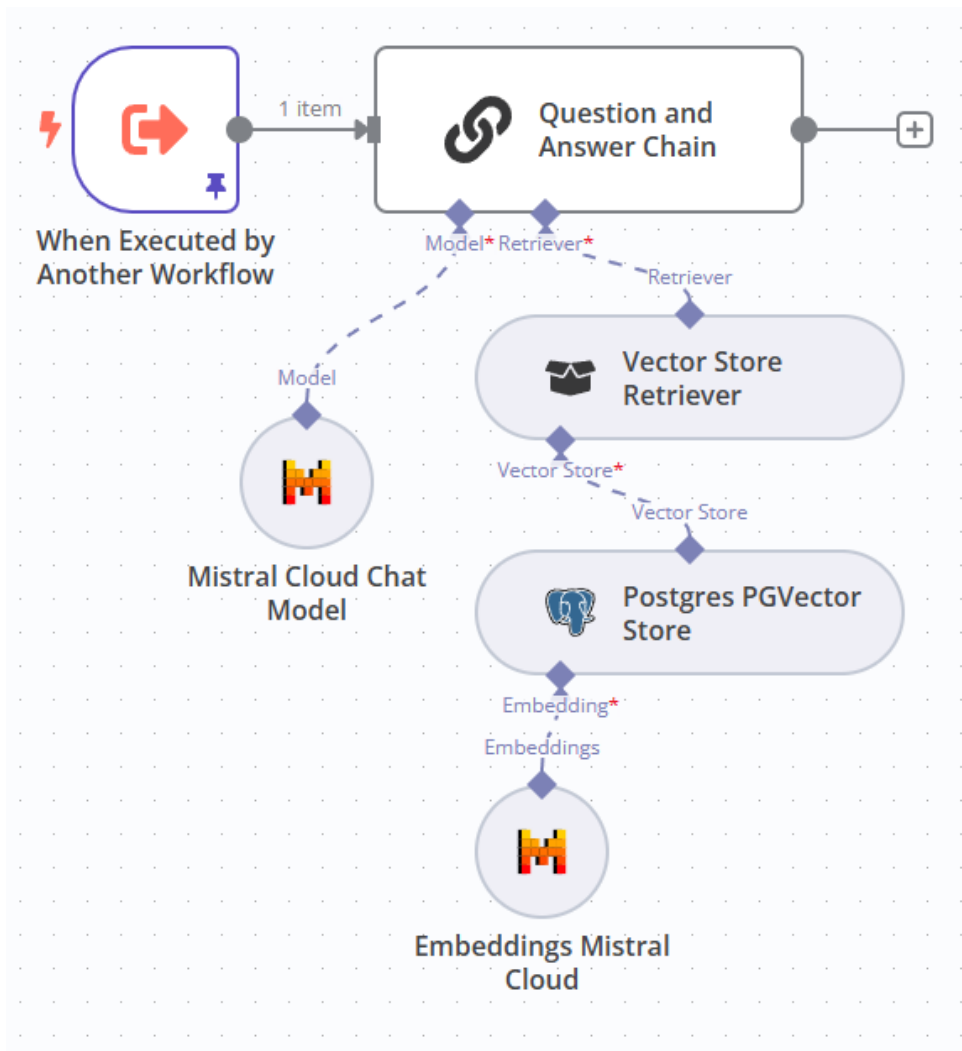
Viimase sammuna on tööriistas veel üks agent, kelle ülesanne on võtta kokku 2-3 lausega iga paberi kokkuvõtted, et anda omakorda lühikokkuvõtte artiklite sisust. Vastuse vormistab ta selliselt, et alguses on paberi ID ja selle all kokkuvõtte paberist (vt Joonis 14). Ka selle agendi mudeliks on *mistral-small* mudel (Mistral AI, 2025). Kui artikkel tundub analüüsi eesmärgi saavutamiseks piisavalt kasulik, siis saab mikrobioloog selle andmebaasist üles otsida kasutades ID-d, alla laadida ning seejärel laadida vektorandmebaasi ülesse, et dokumentidest otsimise tööriistaga selle kohta päringuid esitada. Automaatne terviktekstide alla laadimine ja kokku võtmine ei olnud võimalik, sest artiklid on avaldatud erinevatel lehtedel ning tihti on ka nad maksumüüri taga, mis muudab ühtse lahenduse välja töötamise liiga keerukaks.

5.4.3 Dokumentidest otsimise tööriist

Viimane tööriist, mis superagendi arsenalis on, on dokumentidele päringute esitamise tööriist ehk *PDF-tool*. Tööriist on oma olemuselt lihtne, see koosneb ainult ühest agendist, kelle ülesanne on eelnevalt üles laetud PDF-failidest koosnevast vektorandmebaasist infot leida ning päringule vastata. Selleks kasutatakse n8n poolt loodud sõlme *Question and Answer Chain*²²,

²² <https://docs.n8n.io/integrations/builtin/cluster-nodes/root-nodes/n8n-nodes-langchain.chainretrievalqa/> Vaadatud 29.04.2025

mis on selle ülesande jaoks juba seadistatud. Vastav mudel on *mistral-small* mudel (Mistral AI, 2025), millega on ühendatud tööriistana *pgvector*²³ vektorandmebaas (vt Joonis 15). Andmed on eelnevalt vektoriseeritud, kasutades Mistrali vektoriseerimise mudelit *mistral-embed* (Mistral AI, 2023).



Joonis 15 Dokumentide kohta pärimise tööriist. Mudel hangib *pgvector* vektorandmebaasis oleva teaduskirjanduse, mis on Mistrali mudeli poolt vektoriseeritud.

Tööriist töötab, kasutades allikapõhist genereerimist, ehk RAG-i, mis on lähemalt kirjeldatud peatükis 2.1. Sisendküsimusele leitakse andmebaasist kõige sarnasemad tekstilõigud, misjärel genereerib agent, siinkohal *Question and Answer Chain* sõlm, saadud info põhjal vastuse, mida peamisele agendile ülesvoolu saata.

²³ <https://github.com/pgvector/pgvector> Vaadatud 15.04.2025

Chat Session d4cc5d33bf0047a890bc4d2589ab4c35 ↻ Latest Logs from Master Agent node

What's the role of butyrate in the gut-skin axis?

Butyrate, a short-chain fatty acid (SCFA), plays a pivotal role in the "gut-skin axis," which refers to the bidirectional communication between the gut and the skin. Gut health, influenced by factors such as diet, can impact the skin's allostasis (homeostasis). Butyrate is produced by gut commensal microbes through the fermentation of dietary fiber. It influences the mucosal immune system by increasing Tregs (regulatory T cells) via SCFAs. This immunomodulatory/anti-inflammatory mechanism also exists in the skin, where commensal bacteria such as *Cutibacterium acnes* produce SCFAs.

Topical butyrate administration may have therapeutic potential for inflammatory skin diseases, as it can counteract exaggerated inflammatory responses and maintain a homeostatic state by exerting a down-regulatory function. Beneficial roles of SCFAs, including butyrate, are mediated by direct activation of G-protein coupled receptors (GPRs) and inhibition of histone deacetylase (HDAC), which promote Tregs proliferation involved in

⬆ | Type a message, or press 'up' arrow for previous one ➤

Master Agent

- Postgres Chat Memory
- Mistral Cloud Chat Model1
- PDF tool
- Mistral Cloud Chat Model1
- Postgres Chat Memory

Joonis 16 Näide dokumentide otsimise tööriista vastusest. Päring ja vastus on Coppola jt., 2022 artikli *Potential Clinical Applications of the Postbiotic Butyrate in Human Skin Diseases* kohta.

Tööriist tagastab vastusena kokkuvõtliku ülevaate sisendküsimusele (vt Joonis 16), kasutades otsinguks vektorandmebaasi. Vastused võivad ka olla mitmest andmebaasi üles laetud dokumendist pärit.

Chat Session 8505a4e2d11a4e3a92524285ecec00eb ↻ Latest Logs from Master Agent node

According to Balabanova et al. (2021), which bacterial species are major industrial producers of vitamin B12? Do any of our strains with high CNB12, MeB12, or AdenB12 levels in milk belong to these species? If not which strain belonging to what species produces the highest amount of B12 in milk?

The bacterial species that are major industrial producers of vitamin B12 are *Pseudomonas denitrificans* and *Propionibacterium freudenreichii*, which can produce up to 214.3 mg L⁻¹ and 206.0 mg L⁻¹ of vitamin B12, respectively.

None of our strains with high CNB12, MeB12, or AdenB12 levels in milk belong to the species *P. denitrificans* and *P. freudenreichii*.

The strain belonging to the species *L. lactis* produces the highest amount of B12 in milk (0.9476418012 µM).

⬆ | Type a message, or press 'up' arrow for previous one ➤

Master Agent

- Postgres Chat Memory
- Mistral Cloud Chat Model1
- PDF tool
- Mistral Cloud Chat Model1
- Database tool
- Mistral Cloud Chat Model1
- Database tool
- Mistral Cloud Chat Model1
- Database tool
- Mistral Cloud Chat Model1

Joonis 17 Näide dokumentide tööriista kasutusest koostöös BioCC andmetega, kasutades infot Balabanova jt. (2021) uuringust. Näha on ka kuidas mudel, kasutab lahenduse leidmiseks intensiivselt tööriistu.

Lisaks võimaldab dokumentidest otsimise tööriist kombineerida infot andmebaasis olevate andmetega, võimaldades pärida agendilt komplekssemaid päringuid. Joonis 17 on näha, kuidas mudel esmalt otsib vajalikku infot teadusartiklist ning seejärel teostab mitu päringut andmebaasile, et lõplik vastus vormistada.

5.5 Failide üles laadimise voog

Toetava voona valmis ka failide üles laadimise voog. Voog sisaldab omakorda kahte alamvoogu: esimene neist tegeleb KEGGCharteri (Sequeira jt., 2022) väljunditega, millega tegelemine on täpsemalt kirjeldatud peatükis 4.4, ning teine tegeleb PDF-failide üles laadimisega. Voo näol on tegu kahe kaustaga, mis jälgivad, kas nendesse on uus fail lisandunud. Sellest tulenevalt on voog mõeldud olema taustal alati aktiivne. KEGGCharteri väljundid laetakse ülesse vastava tabeli skeemi järgi (vt Lisa I). PDF-failid laetakse kausta lisamisel üles pgvector vektorandmebaasi. Enne andmebaasi lisamist jupitatakse PDF-fail kuni 3000 tähemärgi suurusteks tükkideks kuni 200-se ülekattega ning muudetakse vektoriteks, kasutades Mistrali *mistral-embed* (Mistral AI, 2023) mudelit.

5.6 Võimalikud edasiarendused

Valminud töövood ja skriptid loodi vastavalt selle projekti käigus esile kerkinud vajadustele. Juhul kui peaks tekkima soov etappide järjestust töövoos muuta või osad etapid üldse välja jätta, siis on võimalik seda teha, kustutades või tõstes ümber n8n-is sõlmede vahelisi lülisid. Töö käigus valmis üks selline alamvoog nimega Krakeni voog, kuid valminud sõlmesid kasutades saab ka muid selliseid alamvooge luua. Lisaks saab töövoogu protsesse vajadusel juurde lisada. Selle jaoks tuleks küll vastavate programmide jaoks uued skriptid teha, kuid kõik praegused skriptid on üles ehitatud sarnaselt, mistõttu saab neid vajadusel aluseks võtta.

Veel saaks edasiarendusena lisada peamisele suhtlusagendile juurde erinevaid tööriistu. Näiteks saaks ühendada lisaks BioCC andmebaasile juurde mingi teise andmebaasi, kust tulemusi otsida. Üks edasiarenduse kohtasid kindlasti on ka andmebaas, millele loodi töö raames skeem, mis rahuldab vaid esmaseid nõudeid agendi töötamiseks.

5.7 Tulemuste valideerimine ja hinnang tööle

Töövoogude töö hindamiseks kasutati testandmeid, millega jooksutati töövoog algusest lõpuni. Seda tehti nii peamise töövoogu kui ka alamtöövoogu puhul. Töövood testiti läbi erinevate parameetri kombinatsioonidega, jättes kord mõned valikulised lahtrid tühjaks ja kord täidetuna. Voost läbitud andmed saadeti tagasi BioCC töötajatele hindamiseks ning nende poolne tagasiside oli positiivne.

Agendi töö hindamiseks kasutati erinevaid juhuslikke sisendeid, mille puhul tulemust oleks võimalik kontrollida. Näiteks dokumentide otsimise tööriista puhul küsiti küsimusi, mille vastuseid oleks võimalik saada ainult andmebaasis olevate dokumentide põhjal. Selline küsimus oli näiteks Joonis 16 nähtav küsimus butüraadi rolli kohta. Vastuse õigsust kontrolliti artiklist endast. Kokkuvõtete tööriista puhul kontrolliti, kas leitud artiklite ID-d vastavad nende artiklitele ning kas leitud artiklid on küsimusele relevantid. Andmebaasi agendi puhul küsiti nii kitsamaid kui laiemaid küsimusi andmebaasis olevate andmete kohta ning kontrolliti, kas agendi poolt esitatud andmed vastuses ka realselt vastavad nendele, mis andmebaasis on.

Töö tulemusena valminud töövood ja suhtlusagent täidavad autori hinnangul põhieesmärgi, milleks on mikrobioloogide tööaja parem optimeerimine. Valminud töövood võimaldavad mikrobioloogil peale parameetrite sisestamist panna voo käima ning minna tegelema muude ülesannetega. Lisaks aja paremale optimeerimisele toob töövoos kasutamine ka üldise ajavõidu, sest töövoos osad käivitatakse automaatselt teineteise järel. Töövoos käivitamisel tekivad programmide tulemused nende vastavatesse alamkaustadesse, mis võimaldab vastava programmi tulemused lihtsalt üles otsida. Valminud suhtlusagent võimaldab aga pärida BioCC enda tootmisandmete kohta ja töövoos käigus tekkinud annoteerimise andmete kohta. Lisaks võimaldab agent vastata teadusartiklite põhjal küsimustele ning otsida ka PubMed andmebaasist uusi artikleid, mida edasi uurida.

Kokkuvõte

Töö raames valmisid automaatsed geeniotsingu töövood ettevõtte BioCC jaoks. Peamine eesmärk oli luua lahendus, mis vähendaks BioCC töötajate ajakulu geeniotsingu teostamisel ning mis võimaldaks töötajatel pühendada aega muudele olulistele asjadele. Veel üks eesmärk oli siduda ettevõtte varasemad tootmisandmed ja info teadusartiklitest analüüsiprotsessiga, võimaldades seeläbi andmete lisaväärindamist.

Voogude valmimine hõlmas endas töövoogude disainimist ja jooksvat väljaarendamist koostöös ettevõttega. Selle jaoks selgitati välja, milline on BioCC-s teostatav geenianalüüsi protsess ning mida iga etapp selles analüüsis endas sisaldab. Esialgsetest mõtetest kujundati koos ettevõttega töövood, mis nende vajadusi rahuldaksid. Lisaks töötati välja ka tehisintellekti roll selles töövoos ning kuidas seda kasutada saaks koos BioCC andmetega, et analüüsiprotsessi täiustada.

Töövoogude kokkupanekuks kasutati automatiseerimise keskkonda n8n. Lisaks loodi ka andmebaasi skeem, mis võimaldaks tehisintellektil BioCC andmetega suhelda. Töövood algavad parameetrite küsimisega, misjärel käivitavad voogude tarbeks loodud skriptid erinevaid töövoos etappe. Töövoos käigus tekkinud genoomi annoteerimise andmed, mis sisaldavad genoomi tootmisinfot, lisatakse andmebaasi, et neid saaks kasutada koostöös BioCC varasemate tootmisandmetega. Töövoos tulemuste ühendamiseks tootmisandmetega loodi suhtlusagent, millel on kasutada vastuste loomiseks kolm tööriista: andmebaasist otsimise tööriist, vektorandmebaasi üles laetud artiklitest otsimise tööriist ning uute teadusartiklite otsimise ning sisu kokku võtmise tööriist.

Agendi ja töövoogude tulemust valideeriti testimisega. Töövoogude puhul jookсутati voost läbi testandmed, mille tulemused saadeti BioCC-le hinnangu andmiseks. Suhtlusagendi tööd kontrolliti, kasutades juhuslikke päringuid, mille vastuste õigsust kontrolliti käsitsi allikatest infot üles otsides. Valideerimise tulemusena leiti, et tehtud töö täidab eesmärgi, võimaldades mikrobioloogidel töövooge kasutades säästa ülesandest sõltuvalt märkimisväärselt aega, kas mõnest tunnist kuni mõne päevani. Lisaks võimaldab loodud suhtlusagent geenianalüüsis andmete lisaväärindamist, ühendades info nii teadusartiklitest, töövoos tulemustest kui ka BioCC tootmisandmetest.

Töövood koos skriptidega on kättesaadavad lehel: <https://github.com/Karlerikkanal/GenomeAnalysisTool>. Tundlik informatsioon on töövoogudest ja agentide süsteemiviipadest eemaldatud.

Viidatud kirjandus

Kanehisa, M., & Goto, S. (2000). KEGG: kyoto encyclopedia of genes and genomes. *Nucleic acids research*, 28(1), 27-30. <https://doi.org/10.1093/nar/28.1.27>

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33, 9459-9474. <https://doi.org/10.48550/arXiv.2005.11401>

Devlin, J., Chang, M. W., Lee, K. & Toutanova, K. (juuni 2019). Bert: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)* (pp. 4171-4186). <https://doi.org/10.48550/arXiv.1810.04805>

Karpukhin, V., Oguz, B., Min, S., Lewis, P. S., Wu, L., Edunov, S., Chen, D. & Yih, W. T. (november 2020). Dense Passage Retrieval for Open-Domain Question Answering. *EMNLP (I)* (pp. 6769-6781). <https://doi.org/10.48550/arXiv.2004.04906>

Nakano, R., Hilton, J., Balaji, S., Wu, J., Ouyang, L., Kim, C., Hesse, C., Jain, S., Kosaraju, V., Saunders, W., Jiang, X., Cobbe, K., Eloundou, T., Krueger, G., Button, K., Knight, M., Chess, B. & Schulman, J. (2022). *Webgpt: Browser-assisted question-answering with human feedback*. arXiv <https://doi.org/10.48550/arXiv.2112.09332>

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., ... Amodei, D. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33, 1877-1901. <https://doi.org/10.48550/arXiv.2005.14165>

Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Hambro, E., Zettlemoyer, L., Cancedda, N. & Scialom, T. (2023). Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36, 68539-68551. <https://doi.org/10.48550/arXiv.2302.04761>

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., & Cao, Y. (jaanuar 2023). ReAct: Synergizing reasoning and acting in language models. *International Conference on Learning Representations (ICLR)*. <https://doi.org/10.48550/arXiv.2210.03629>

- Shen, Y., Song, K., Tan, X., Li, D., Lu, W. & Zhuang, Y. (2023). HuggingGPT: Solving AI Tasks with ChatGPT and its Friends in Hugging Face. *Advances in Neural Information Processing Systems*, 36, 38154-38180. <https://doi.org/10.48550/arXiv.2410.21276>
- Wu, Q., Bansal, G., Zhang, J., Wu, Y., Li, B., Zhu, E., Jiang, L., Zhang, X., Zhang, S., Liu, J., Awadallah, A. H., White, R. W., Burger, D. & Wang, C. (2023). Autogen: Enabling next-gen llm applications via multi-agent conversation. *arXiv preprint arXiv:2308.08155*. <https://doi.org/10.48550/arXiv.2308.08155>
- Krueger, F. (i.a). *Trim Galore!* Github. Kasutatud 10.02.2025 <https://github.com/FelixKrueger/TrimGalore>
- Martin, M. (2011). Cutadapt removes adapter sequences from high-throughput sequencing reads. *EMBnet. journal*, 17(1), 10-12. <https://doi.org/10.14806/ej.17.1.200>
- Andrews, S. (2010). FastQC: A Quality Control Tool for High Throughput Sequence Data. Kasutatud 10.02.2025 <http://www.bioinformatics.babraham.ac.uk/projects/fastqc/>
- Li, D., Liu, C. M., Luo, R., Sadakane, K., & Lam, T. W. (2015). MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph. *Bioinformatics*, 31(10), 1674-1676. <https://doi.org/10.1093/bioinformatics/btv033>
- Bowe, A., Onodera, T., Sadakane, K., & Shibuya, T. (september 2012). Succinct de Bruijn graphs. *International workshop on algorithms in bioinformatics* (pp. 225-235). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-33122-0_18
- Bankevich, A., Nurk, S., Antipov, D., Gurevich, A. A., Dvorkin, M., Kulikov, A. S., Lesin, V. M., Nikolenko, S. I., Pham, S., Prjibelski, A. D., Pyshkin, A. V., Sirotkin, A. V., Vyahhi, N., Tesler, G., Alekseyev, M. A., & Pevzner, P. A. (2012). SPAdes: A new genome assembly algorithm and its applications to single-cell sequencing. *Journal of Computational Biology*, 19(5), 455–477. <https://doi.org/10.1089/cmb.2012.0021>
- Gurevich, A., Saveliev, V., Vyahhi, N., & Tesler, G. (2013). QUAST: quality assessment tool for genome assemblies. *Bioinformatics*, 29(8), 1072-1075. <https://doi.org/10.1093/bioinformatics/btt086>
- Seemann T. (i.a). *ABRicate*, Github. Kasutatud 10.02.2025 <https://github.com/tseemann/abricate>

Feldgarden, M., Brover, V., Haft, D. H., Prasad, A. B., Slotta, D. J., Tolstoy, I., Tyson, G. H., Zhao, S., Hsu, C. H., McDermott, P. F., Tadesse, D. A., Morales, C., Simmons, M., Tillman, G., Wasilenko, J., Folster, J. P., & Klimke, W. (2019). Validating the AMRFinder Tool and Resistance Gene Database by Using Antimicrobial Resistance Genotype-Phenotype Correlations in a Collection of Isolates. *Antimicrobial agents and chemotherapy*, *63*(11), 10-1128. <https://doi.org/10.1128/AAC.00483-19>

Jia, B., Raphenya, A. R., Alcock, B., Waglechner, N., Guo, P., Tsang, K. K., Lago, B. A., Dave, B. M., Pereira, S., Sharma, A. N., Doshi, S., Courtot, M., Lo, R., Williams, L. E., Frye, J. G., Elsayegh, T., Sardar, D., Westman, E. L., Pawlowski, A. C., Johnson, T. A., Brinkman, F. S. L., Wright, G. D., McArthur, A. G. (2017). CARD 2017: expansion and model-centric curation of the comprehensive antibiotic resistance database. *Nucleic acids research*, *45*(D1), D566–D573. <https://doi.org/10.1093/nar/gkw1004>

Zankari, E., Hasman, H., Cosentino, S., Vestergaard, M., Rasmussen, S., Lund, O., Aarestrup, F. M., & Larsen, M. V. (2012). Identification of acquired antimicrobial resistance genes. *The Journal of antimicrobial chemotherapy*, *67*(11), 2640–2644. <https://doi.org/10.1093/jac/dks261>

Gupta, S. K., Padmanabhan, B. R., Diene, S. M., Lopez-Rojas, R., Kempf, M., Landraud, L., & Rolain, J. M. (2014). ARG-ANNOT, a new bioinformatic tool to discover antibiotic resistance genes in bacterial genomes. *Antimicrobial agents and chemotherapy*, *58*(1), 212–220. <https://doi.org/10.1128/AAC.01310-13>

Chen, L., Zheng, D., Liu, B., Yang, J., & Jin, Q. (2016). VFDB 2016: hierarchical and refined dataset for big data analysis--10 years on. *Nucleic acids research*, *44*(D1), D694–D697. <https://doi.org/10.1093/nar/gkv1239>

Carattoli, A., Zankari, E., García-Fernández, A., Voldby Larsen, M., Lund, O., Villa, L., Møller Aarestrup, F., & Hasman, H. (2014). In silico detection and typing of plasmids using PlasmidFinder and plasmid multilocus sequence typing. *Antimicrobial agents and chemotherapy*, *58*(7), 3895–3903. <https://doi.org/10.1128/AAC.02412-14>

Ingle, D. J., Valcanis, M., Kuzevski, A., Tauschek, M., Inouye, M., Stinear, T., Levine, M. M., Robins-Browne, R. M., & Holt, K. E. (2016). In silico serotyping of E. coli from short read data identifies limited novel O-loci but extensive diversity of O:H serotype combinations

within and between pathogenic lineages. *Microbial genomics*, 2(7), e000064.

<https://doi.org/10.1099/mgen.0.000064>

Doster, E., Lakin, S. M., Dean, C. J., Wolfe, C., Young, J. G., Boucher, C., Belk, K. E., Noyes, N. R. & Morley, P. S. (2020). MEGARes 2.0: a database for classification of antimicrobial drug, biocide and metal resistance determinants in metagenomic sequence data. *Nucleic acids research*, 48(D1), D561-D569. <https://doi.org/10.1093/nar/gkz1010>

Gillings, M. R. (2014). Integrons: Past, present, and future. *Microbiology and Molecular Biology Reviews*, 78(2), 257–277. <https://doi.org/10.1128/mnbr.00056-13>

Néron, B., Littner, E., Haudiquet, M., Perrin, A., Cury, J., & Rocha, E. P. (2022). Integron-Finder 2.0: identification and analysis of integrons across bacteria, with a focus on antibiotic resistance in *Klebsiella*. *Microorganisms*, 10(4), 700. <https://doi.org/10.3390/microorganisms10040700>

Nawrocki, E. P., & Eddy, S. R. (2013). Infernal 1.1: 100-fold faster RNA homology searches. *Bioinformatics*, 29(22), 2933-2935. <https://doi.org/10.1093/bioinformatics/btt509>

Eddy, S. R. (2011). Accelerated profile HMM searches. *PLoS computational biology*, 7(10), e1002195. <https://doi.org/10.1371/journal.pcbi.1002195>

Hyatt, D., Chen, G. L., LoCascio, P. F., Land, M. L., Larimer, F. W., & Hauser, L. J. (2010). Prodigal: prokaryotic gene recognition and translation initiation site identification. *BMC bioinformatics*, 11, 1-11. <https://doi.org/10.1186/1471-2105-11-119>

Xie, Z., & Tang, H. (2017). ISEScan: automated identification of insertion sequence elements in prokaryotic genomes. *Bioinformatics*, 33(21), 3340-3347. <https://doi.org/10.1093/bioinformatics/btx433>

Wood, D. E., Lu, J., Langmead, B. (2019). Improved metagenomic analysis with Kraken 2. *Genome Biology*, 20, 257. <https://doi.org/10.1186/s13059-019-1891-0>

Lu, J., Rincon, N., Wood, D. E., Breitwieser, F. P., Pockrandt, C., Langmead, B., Salzberg, S. L., & Steinegger, M. (2022). Metagenome analysis using the Kraken software suite. *Nature Protocols*, 17, 2815–2839. <https://doi.org/10.1038/s41596-022-00738-y>

Lu, J., Breitwieser, F. P., Thielen, P., & Salzberg, S. L. (2017). Bracken: estimating species abundance in metagenomics data. *PeerJ Computer Science*, 3, e104. <https://doi.org/10.7717/peerj-cs.104>

- Seemann, T. (2014). Prokka: rapid prokaryotic genome annotation. *Bioinformatics*, 30(14), 2068-2069. <https://doi.org/10.1093/bioinformatics/btu153>
- Sequeira, J. C., Rocha, M., Alves, M. M., & Salvador, A. F. (2022). UPIMAPI, reCOGnizer and KEGGCharter: Bioinformatics tools for functional annotation and visualization of (meta)-omics datasets. *Computational and Structural Biotechnology Journal*, 20, 1798-1810. <https://doi.org/10.1016/j.csbj.2022.03.042>
- Page, A. J., Cummins, C. A., Hunt, M., Wong, V. K., Reuter, S., Holden, M. T., Fookes, M., Falush, D., Keane, J. A. & Parkhill, J. (2015). Roary: rapid large-scale prokaryote pan genome analysis. *Bioinformatics*, 31(22), 3691-3693. <https://doi.org/10.1093/bioinformatics/btv421>
- Tange, O. (2023, 22. november). *GNU Parallel 20231122 ('Grindavik')* [Tarkvara]. Zenodo. <https://doi.org/10.5281/zenodo.10199085>
- OpenAI. (2024). *GPT-4o* (Mai 2024 versioon) [suur keelemudel]. <https://openai.com>
- Mistral AI. (2024, 18. november). *Mistral Large 24.11* [suur keelemudel]. <https://mistral.ai>
- Mistral AI. (2025, 13. jaanuar). *Codestral 25.01* [koodi genereerimise mudel]. <https://mistral.ai>
- Mistral AI. (2025, 17. märts). *Mistral Small 25.03* [keelemudel]. <https://mistral.ai>
- Mistral AI (detsember 2023). *Mistral Embed 23.12* [vektoriseerimise mudel]. <https://mistral.ai>
- Sayers, E. (2009, 26. mai). A General Introduction to the E-utilities. Entrez Programming Utilities Help [Internet]. *Bethesda (MD): National Center for Biotechnology Information (US)*. <https://www.ncbi.nlm.nih.gov/books/NBK25497/> Vaadatud 14.05.2025
- Coppola, S., Avagliano, C., Sacchi, A., Laneri, S., Calignano, A., Voto, L., Luzzeti, A. & Berni Canani, R. (2022). Potential clinical applications of the postbiotic butyrate in human skin diseases. *Molecules*, 27(6), 1849. <https://doi.org/10.3390/molecules27061849>
- Balabanova, L., Averianova, L., Marchenok, M., Son, O., & Tekutyeva, L. (2021). Microbial and genetic resources for cobalamin (vitamin B12) biosynthesis: From ecosystems to industrial biotechnology. *International Journal of Molecular Sciences*, 22(9), 4522. <https://doi.org/10.3390/ijms22094522>

Lisad

I. Annotatsioonide tabeli skeem

```
1. CREATE TABLE kegg_annotations (  
2.     id                                INTEGER,  
3.     "Strain number"                   TEXT,  
4.     locus_tag                         TEXT,  
5.     ftype                             TEXT,  
6.     length_bp                        INTEGER,  
7.     gene                              TEXT,  
8.     ec_number                         TEXT,  
9.     cog                               TEXT,  
10.    product                           TEXT,  
11.    quantification_keggcharter       INTEGER,  
12.    taxon_keggcharter                 TEXT,  
13.    ko_ec_column                      TEXT,  
14.    ec_ec_column                      TEXT,  
15.    ko_cog_column                     TEXT,  
16.    ko_keggcharter                    TEXT,  
17.    ec_number_keggcharter             TEXT  
18. );  
19.
```

II. Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, **Karl-Erik Kanal**

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose **Genoomiotsingu töövoogude arendamine BioCC OÜ jaoks**, mille juhendaja on **Martin Rebane**, reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada Tartu Ülikooli digitaalarhiivi kuni autoriõiguse kehtivuse lõppemiseni;
2. annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi kaudu Creative Commons'i litsentsiga CC BY NC ND 4.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni;
3. olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile;
4. kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

Karl-Erik Kanal

15.05.2025