



Kustomoitu GPT: Luo, ohjeista, tehosta

Kuinka kehotteet muokkaavat kielimallien vastauksia

Matias Karmanto

OPINNÄYTETYÖ
Huhtikuu 2024

Tietojenkäsittelyn tutkinto-ohjelma
Ohjelmistotuotanto

TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Tietojenkäsittelyn tutkinto-ohjelma
Ohjelmistotuotanto

KARMANTO, MATIAS:
Kustomoitu GPT: Luo, ohjeista, tehosta
Kuinka kehotteet muokkaavat kielimallien vastauksia

Opinnäytetyö 41 sivua
Huhtikuu 2024

Opinnäytetyössä keskityttiin tutkimaan ja kehittämään omaa generatiivista esikoulutettua muunninta (GPT) hyödyntäen kehotteiden teoriaa. Työn tavoitteena oli selvittää, kuinka kehotteiden avulla voidaan ohjata kielimallia tuottamaan haluttuja vastauksia ja parantaa sen suorituskykyä. Opinnäytetyö aloitettiin syksyllä 2023, jolloin tekoäly-yhtiö OpenAI julkisti GPT-kauppapaikan ja mahdollisti käyttäjille omien GPT-mallien luomisen. Työssä kehitettiin oma GPT-malli, Expenses-GPT, joka osoitti, kuinka tekoälyn integroiminen ulkoisiin palveluihin voi tuoda uusia käyttömahdollisuuksia.

Opinnäytetyön alkuosassa käsiteltiin erilaisia kielimalleja sekä niiden arkkitehtuuria ja syvennyttiin kehotteiden teoriaan. Työssä esiteltiin myös kehotteiden suunnitteluun ja hienosäätöön liittyviä parhaita käytäntöjä. Tutkimuksessa kehitetty GPT-malli demonstroi, miten kehotteita voidaan käyttää tekoälyn ohjaamiseen ja kuinka tämä mahdollistaa tarkempien ja kontekstuaalisesti relevanttien vastausten tuottamisen. Mallin kehitysprosessi ja sen integrointi käyttämään olemassa olevaa rajapintaa kuvattiin yksityiskohtaisesti.

Opinnäytetyössä ei hyödynnetty OpenAI Assistant -rajapintaa rajallisen laajuuden vuoksi, vaan keskityttiin perusteellisesti kehotteiden teoriaan ja sen soveltamiseen käytännössä. Työssä käytetty aineisto koostui OpenAI:n dokumentaatiosta, luotettavista verkkosivustoista ja tieteellisistä julkaisuista, mikä vahvisti tutkimuksen tieteellistä luotettavuutta.

Tutkimus osoitti, että GPT-mallien kehittäminen ja niiden ohjaaminen kehotteiden avulla tarjoaa merkittäviä mahdollisuuksia tekoälyn soveltamiseen erilaisissa konteksteissa. Työ tarjosi arvokasta tietoa kehotteiden teorian soveltamisesta käytännössä ja demonstroi, kuinka tekoälyn avulla voidaan saavuttaa konkreettisia hyötyjä. Jatkokehitysideana esitettiin omien GPT-mallien integrointi esimerkiksi verkkosivuille tai sovelluksiin tuoden niille lisäarvoa tekoälyn avulla.

Asiasanat: GPT, ChatGPT, kehotteet, kehotteen muotoilu, sovelluskehitys

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Degree Programme in Business Information Systems
Software Production

KARMANTO, MATIAS:
Customizing GPT: Create, Instruct, Enhance
How Prompts Shape the Output of the LLMs

Bachelor's thesis 41 pages
April 2024

This thesis explored prompt engineering for developing a custom GPT model, specifically focusing on the development of "Expenses-GPT". It investigated how tailored prompts can enhance the performance of ChatGPT-based models for specific tasks. The project demonstrated the utility of integrating external services through API interfaces to extend capabilities of GPT.

The methodology consisted of an exploration of language models, with a focus on GPT-4-turbo and the formulation of effective prompts. This process involved crafting a custom GPT and examining strategies for influencing its output. The research delved into the theory of prompts and culminated in the development of a GPT model.

Key findings highlight the ability of custom GPT model to generate relevant and contextually appropriate responses, demonstrating the practical benefits of integrating GPT with external services for enhanced functionality.

The conclusions drawn from this study underscore the effectiveness of prompt engineering in refining the utility of GPT models for specialised tasks. Future development could involve integrating the custom GPT into web services to provide additional value to users.

Key words: prompts, prompt engineering, GPT, LLM

SISÄLLYS

1	JOHDANTO	6
1.1	Opinnäytetyön aihe, kulku ja tarkoitus	6
1.2	Opinnäytetyön rajoituksia	7
2	OPINNÄYTETYÖN TOTEUTUS	8
2.1	Opinnäytetyössä käytetty aineisto	8
2.2	Aikaisempia toteutuksia Suomessa	8
3	GENERATIIVINEN ENNALTA KOULUTETTU MUUNNIN (GPT)	10
3.1	Johdanto kielimalleihin	10
3.2	Koulutusprosessi	10
3.3	Sovellusalueet	11
4	GPT JA KEHOTTEET	13
4.1	Kehotteen muotoilun perusteet	13
4.2	Kehotteen osat	14
4.3	Tokenit	14
4.4	Hyviä käytäntöjä kehotteen rakentamiseen	16
4.5	Kehoteet	17
4.5.1	Zero-shot	17
4.5.2	Few-shot	19
4.5.3	Chain-of-thought	20
5	OMAN GPT:N LUONTI	22
5.1	GPT Builder ja Editor	23
5.2	Kehotteiden luonti	24
5.2.1	Rooli	25
5.2.2	Vaiheittainen prosessi	25
5.2.3	Rajoitteet ja esimerkki	26
5.3	GPT:n toiminnallisuudet	27
6	GPT ACTIONS JA INTEGROINTI API-RAJAPINTAAN	29
7	POHDINTA	36
7.1	GPT:t ja niiden kyvykkyydet	36
7.2	Kielimallien ja GPT:n haasteet	37
7.3	Opinnäytetyön luotettavuus	38
7.4	Opinnäytetyön tuotosten tarkastelu ja jatkokehitysideat	38
	LÄHTEET	40

LYHENTEET JA TERMIT

Assistant API	OpenAI:n rajapinta, jolla ohjeistaa GPT:tä
BERT	Googlen kielimalli
Capabilities	GPT:n erikoiskyvykkyudet
ChatBot	Keskustelurobotti
ChatGPT	OpenAI:n keskustelurobotti
Code-interpreter	Kooditulkki
DALL-E	OpenAI:n kuviengenerointitekoäly
Gemini	Googlen keskustelurobotti
GPT	Generative Pre-trained Transformer – Generatiivinen esikoulutettu muunnin
GPT Actions	GPT:lle määriteltävät toiminnot
GPT-3.5 / GPT-4 / GPT4-turbo	GPT-versioita
GPT-Store	Sovelluskauppa GPT:eille
HTTP	Hypertext Transfer Protocol
LLM	Large Language Models (suuret kielimallit)
NLP	Natural Language Processing
OpenAI	Tekoäly-yhtiö
OpenAPI-spesifikaatio	Standardisoitu tapa dokumentoida rajapinnan toimintoja
Python	Ohjelmointikieli
Token	Sana, sananosa tai erikoismerkki
Tokenisointi	Tekstin jakaminen pienempiin yksiköihin, tokeneihin
Transformer-arkkitehtuuri	Syväoppimisen malli, johon GPT perustuu

1 JOHDANTO

1.1 Opinnäytetyön aihe, kulku ja tarkoitus

OpenAI julkisti syksyllä 2023 mahdollisuuden käyttäjille luoda omia generatiivisia muuntimia (GPT). Nämä GPT:t toimivat tekoäly-yhtiö OpenAI ChatGPT käyttöliittymällä, mutta niitä voidaan ohjeistaa yksityiskohtaisesti ja täten määrittää kuinka GPT vastaa, määrittää materiaali mistä tietoa haetaan ja yhdistää ulkopuolisia palveluita GPT:n käyttöön rajapintojen kautta. Nämä ominaisuudet sekä yleinen teoria kielimallien ja GPT:n ympärillä on laaja, mutta mielenkiintoinen aihe, ja opinnäytetyö pyrkii tutkimaan, kuinka GPT:n kaltaisista kielimalleista saa parempia tehtävän suorittajia.

Opinnäytetyön aiheeksi on valittu kielimallit, erityisesti GPT-4-turbo, ja kehoitteiden eli ohjeiden luominen tälle kielimallille. Opinnäytetyössä keskitytään luomaan oma GPT ja tutkimaan, kuinka käyttäjä voi ohjata kielimallia ja sen vastauksia haluamaansa suuntaan. Opinnäytetyössä tullaan käymään kehoitteiden teoriaa läpi sekä ottamaan esimerkkejä muutamista hyväksi todetuista kehotetekniikoista.

Opinnäytetyön alussa käsitellään erilaisia kielimalleja ja niiden arkkitehtuuria. Tutkimus syventyy tämän jälkeen kehoitteiden teoriaan, parhaisiin käytäntöihin ja tutustuu muutamiin usein käytettyihin tekniikoihin. Prosessin loppuvaiheessa kehitetään oma GPT-malli, jolle laaditaan ohjeistus ja integroidaan se käyttämään olemassa olevaa rajapintaa, jonka kautta GPT voi tehdä muutoksia tietokantaan. Vaikka lopullinen GPT-tuotos toimii kyvykkyyksien demonstraationa, ei sille ole suunniteltu varsinaista käyttötarkoitusta. Pohdintaosiossa pyritään hakemaan erilaisia sovellusalueita kielimalleille ja tarkastelemaan niiden ongelmia sekä haasteita.

1.2 Opinnäytetyön rajauksia

Opinnäytetyössä ei tulla hyödyntämään OpenAI Assistant APIa, joka mahdollistaisi GPT-mallin ohjelmallisen ohjaamisen ja sen integroinnin ulkoisiin palveluihin. Vaikka kyseisen rajapinnan käyttö tarjoaisi lisäasetusten määrittelyn mahdollisuuden, opinnäytetyön rajallisen laajuuden vuoksi tällaisten ominaisuuksien hyödyntäminen jätetään mahdollisen jatkokehityksen varaan. Lisäksi kustomoidun GPT:n luomiseen ja käyttöön on välttämätöntä hankkia GPT-plus-jäsenyys, ja OpenAI Assistant API:n käyttöön liittyvät kustannukset ovat kutsukohtaisia.

Opinnäytetyössä tarkastellaan tekoälyä ja kielimalleja yleisellä tasolla. Vaikka kehoitteiden teoriaa käydään läpi huomattavasti syvällisemmin, tämäkin aihealue on laaja, ja uutta teoriaa julkaistaan jatkuvasti. Siksi keskitytään yleisimpiin käyttötapauksiin ja tarjotaan lukijalle ohjeita helposti omaksuttaviin kehotetekniikoihin. Tämä rajaus mahdollistaa keskittymisen oleelliseen ja tarjoaa perustan mahdollisille tuleville syventäville tutkimuksille.

2 OPINNÄYTETYÖN TOTEUTUS

2.1 Opinnäytetyössä käytetty aineisto

Opinnäytetyössä käytetty aineisto koostuu monipuolisesti eri lähteistä, mukaan lukien OpenAI:n tarjoama dokumentaatio, eri alojen asiantuntijoiden ylläpitämät luotettavat verkkosivustot sekä laaja valikoima tieteellisiä julkaisuja. OpenAI:n dokumentaatio on tarjonnut arvokasta tietoa suoraan teknologian kehittäjiltä tarjoten näkemyksiä ja ohjeistuksia käytettyjen tekoälymallien ominaisuuksista ja käyttötapauksista. Verkkosivustot, jotka on tunnistettu luotettaviksi tiedonlähteiksi, ovat mahdollistaneet ajankohtaisen ja relevantin tiedon keräämisen, tarjoten syvällisempää ymmärrystä aiheesta. Tieteelliset julkaisut puolestaan ovat tuoneet opinnäytetyöhön vankkaa tutkimuspohjaa esitellen alalla tehtyä edistystä, kriittisiä analyysejä ja uusia näkökulmia, jotka ovat olennaisia opinnäytetyön tavoitteiden saavuttamiseksi. Näiden lähteiden monipuolinen hyödyntäminen on mahdollistanut kattavan ja monitahoisen näkemyksen aiheeseen, vahvistaen opinnäytetyön tieteellistä luotettavuutta.

2.2 Aikaisempia toteutuksia Suomessa

Suomessa on toteutettu useita opinnäytetöitä GPT-mallien soveltamisesta ohjelmistokehitykseen ja niiden vaikutuksiin alalla. Lähteen (2023) opinnäytetyö keskittyy GPT-mallien tuomiin potentiaalsiin hyötyihin, haasteisiin ja vaikutuksiin ohjelmistokehityksen eri osa-alueilla, käyttäen tutkimusmenetelminä kirjallisuuskatsausta ja tapaustutkimuksia. Tulokset osoittavat, että GPT-mallit voivat tarjota merkittäviä etuja ohjelmistokehityksessä.

Niinivaara (2023) puolestaan tutkii ChatGPT:n ja muiden GPT-pohjaisten sovellusten hyödyntämistä ohjelmistokehityksessä perehtyen tekoälyn historiaan ja GPT-kielimalliin tarkemmin. Niinivaaran (2023) mukaan Tampereen ammattikorkeakoulun tietotekniikan opiskelijoiden haastatteluiden perusteella tekoälyn käyttöön suhtaudutaan positiivisesti, ja ChatGPT nähdään tehokkaana työkaluna ohjelmistokehityksessä.

Sekä Lähteen (2023) että Niinivaaran (2023) tutkimukset korostavat GPT-mallien hyötyjä, kuten automaattista koodin generointia ja bugien tunnistusta, mutta myös haasteita, kuten väärinkäytön mahdollisuutta ja työllisyyden vähenemistä joillakin aloilla. Niinivaaran (2023) työssä korostetaan lisäksi tekoälytyökalujen eettisen käytön tärkeyttä.

Buruk (2023) tutkii GPT-3.5:n käyttöä akateemisessa kirjoittamisessa, esitellen kuusi eri menetelmää sen hyödyntämiseen. Tutkimuksessa käsitellään GPT-3.5:n etuja, kuten kirjoittamisen tehokkuuden parantamista, ja haasteita, kuten plagiointia ja väärän tiedon levittämistä. Buruk (2023) korostaa GPT-3.5-mallien läpinäkyvän käytön tarvetta akateemisessa kirjoittamisessa ja analysoi henkilökohtaisen kokemuksensa perusteella saatuja tuloksia ja niiden vaikutusta kirjoittamisen etiikkaan.

3 GENERATIIVINEN ENNALTA KOULUTETTU MUUNNIN (GPT)

3.1 Johdanto kielimalleihin

Viime vuosina kielimallien kehityksessä on otettu huomattavia edistysaskeleita, joista erityisesti suurten kielimallien (LLMs) kehitys on noussut merkittävään asemaan. Näiden edistyneiden mallien, kuten OpenAI:n GPT-sarjan ja Googlen BERTin, arvostus perustuu niiden kykyyn prosessoida, ymmärtää ja tuottaa luonnollista kieltä sekä suoriutua laajasta valikoimasta tehtäviä. Tähän kuuluvat muun muassa kysymyksiin vastaaminen, tekstien tiivistäminen ja luovan sisällön tuottaminen. Mallit hyödyntävät transformer-arkkitehtuuria ja niiden koulutuksessa käytetään suuria datamääriä. Tämä mahdollistaa näennäisesti kielten syvällisen ymmärtämisen ja monimutkaisten kielellisten rakenteiden hallinnan. Suurten kielimallien kyky mukautua erilaisiin käyttöyhteyksiin tekee niistä arvokkaita työkaluja sekä akateemisessa tutkimuksessa että liiketoiminnassa. Ne ovat olennaisesti muuttaneet tapamme käsitellä kieltä ja informaatiota digitaalisessa ympäristössä. (What are LLMs? n.d.)

3.2 Koulutusprosessi

Suurten kielimallien kehittäminen etenee neljän päävaiheen kautta: aineiston kerääminen ja esivalmistelu, mallin asetusten määrittely, varsinaisen koulutusjakson läpivienti, sekä erityiskohtiin keskittyvä hienosäätö. Prosessi alkaa hankkimalla laajoja ja monimuotoisia tekstiaineistoja, jotka valmistellaan mallin opettamista varten. Asetusvaiheessa kielimallin perusrakenteet, kuten eri kerrosten lukumäärä, määritetään. Koulutuksen aikana malli harjoitetaan ennakoimaan tekstijatkumia. Hienosäätövaiheessa mallia optimoidaan spesifejä käyttötarkoituksia varten, ja tässä yhteydessä voidaan myös hyödyntää ihmisen antamaa palautetta mallin vastausten laadun kehittämiseksi. (How to Overcome LLM... n.d.)

Suurten kielimallien koulutusprosessissa hyödynnetään useita edistyneitä menetelmiä, joilla pyritään optimoimaan mallin suorituskykyä ja vähentämään

koulutusdatan sisältämiä ei-toivottuja ennakkoluuloja. Näitä menetelmiä ovat muun muassa kehotteen suunnittelu (prompt engineering), kehotteen hienosäätö (prompt-tuning) ja vahvistusoppiminen ihmispalautteen perusteella (RLHF). (What are LLMs? n.d.)

Kehotteen suunnittelussa luodaan tarkoin muotoiltuja kehoitteita tai vihjeitä, jotka ohjaavat tekoälymallia tuottamaan halutunlaisia vastauksia. Tämä menetelmä auttaa ohjaamaan mallia tuottamaan tarkempia ja relevantimpia vastauksia tietyissä tehtävissä tai konteksteissa.

Kehotteen hienosäädöllä pyritään säätämään näitä kehoitteita vielä entistä tarkemmin, jotta mallin suorituskyky tietyssä tehtävässä tai kontekstissa paranisi. Tämä menetelmä mahdollistaa mallin sopeutumisen erilaisiin tehtäviin vähäisellä koulutusdatan määrällä.

Vahvistusoppiminen ihmispalautteen perusteella on edistyneempi tekniikka, jossa mallia koulutetaan ihmisten antaman positiivisen ja negatiivisen palautteen avulla. Tämä menetelmä on erityisen hyödyllinen ei-toivottujen harhaluulojen tunnistamisessa ja korjaamisessa koulutusdatassa. Ihmispalaute auttaa ohjaamaan mallin oppimisprosessia suotuisampaan suuntaan parantaen mallin kykyä tuottaa eettisesti ja faktuaalisesti korrekteja vastauksia. (What are LLMs? n.d.)

3.3 Sovellusalueet

Suuret kielimallit ovat tuoneet merkittäviä uudistuksia monille toimialoille tarjoamalla kehittyneitä ratkaisuja erilaisiin haasteisiin. Niiden vahvuus luonnollisen kielen ymmärryksessä ja käsittelyssä mahdollistaa käytön esimerkiksi chatbottien, virtuaaliassistenttien, sisällön tuotannon ja käännösten parissa. Chatbotit ja virtuaaliassistentit käyttävät näiden mallien kykyä luoda sujuvia ja kontekstiin sopivia vastauksia, mikä nostaa asiakaspalvelun laatua ja nopeutta. (What are LLMs? n.d.)

Sisällöntuotannossa nämä mallit helpottavat blogikirjoitusten, markkinointiaineiston ja muun tekstisisällön automaattista tuottamista. Tutkimuksessa ne auttavat purkamaan ja analysoimaan suuria datamääriä tehostaen näin tiedonkeruuta ja päätöksentekoa. Kielten kääntämisessä suuret kielimallit mahdollistavat tarkat ja asiayhteyteen sopivat käännökset, mikä auttaa voittamaan kielellisiä esteitä ja edistää kansainvälistä yhteisymmärrystä. (What are LLMs? n.d.)

4 GPT JA KEHOTTEET

4.1 Kehotteen muotoilun perusteet

Jotta yleiskäyttöisiä suuria kielimalleja voidaan ohjeistaa suorittamaan tiettyjä tehtäviä, käyttäjien on annettava niille kehotteita – tekstiohjeita ja esimerkkejä halutuista tuloksista. Nämä kehotteet toimivat taustakontekstina ja oppaana kielimalleille, jotta ne voivat tuottaa tekstiä, joka vastaa käyttäjien tavoitteita.

Kehotteen muotoilu on tärkeä osa tehokkaan vuorovaikutuksen luomista tekoälymallien kanssa. Kuten Lane, Hapke & Howard (2019, luku 1) korostavat, on ensiarvoisen tärkeää ymmärtää, miten koneet analysoivat ja tuottavat tekstiä. Tämä ymmärrys auttaa meitä luomaan kehotteita, jotka ohjaavat tekoälymallia tuottamaan halutunlaisia vastauksia.

Tehokkaan kehotteen luominen ei ole pelkästään oikeiden sanojen valitsemista, vaan se edellyttää syvällistä vuoropuhelua ihmisen ja koneen välillä, tähdäten yhteiseen ymmärrykseen tehtävästä ja tavoitteista. Kehotteen on välitettävä selkeästi konteksti, toivotut lähtöparametrit ja mahdolliset rajoitukset samalla kun se ohjeistaa tekoälyä käyttämään luovuuttaan ja älykkyyttään vastausten tuottamiseen.

Kuten Wang, Chakravarthy, Munechika & Chau (2024) ovat havainneet kehotteiden suunnittelu kielimalleille tuo esiin useita haasteita. Vaikka kehotteen antaminen vaikuttaisi olevan yksinkertainen tehtävä, samanlainen kuin ihmisen ohjeistaminen, tehokkaiden ja yleispätevien kehotestrategioiden kehittäminen on itse asiassa vaativaa. Käyttäjät joutuvat usein turvautumaan kokeiluun ja virheisiin ja käyttämään epäintuitiivisia menetelmiä kuten ”ajattele askel askeleelta” -tyyppisiä ohjeita kehotteissaan, mikä korostaa ohjeistamisen monimutkaisuutta kielimalleille.

Zamfirescu-Pereira, Wong, Hartmann & Yang (2023) korostavat tutkimuksessaan, että kehotteen tai kehotestrategian vaikutus kielimallin tuottamiin tuloksiin ja tapa, jolla kehotteet uudelleenmuotoilevat kielimallien

miljardeja parametrejä uudelleen koulutuksen aikana, ovat molemmat keskeisiä tutkimuskohteita luonnollisen kielen käsittelyn (Natural Language Processing, NLP) alalla. He huomauttavat myös, että luotettavien kehotteiden suunnittelun prosessien puute haastaa alalla työskenteleviä asiantuntijoita. Zamfirescu-Pereiran ym. (2023) mukaan NLP-asiantuntijat käyttävät laajasti kokeiluja ja virheiden kautta oppimista iteratiivisesti testaten ja arvioiden eri kehotestrategioiden vaikutuksia konkreettisiin syöte-tulospareihin, systemaattinen arviointi suurilla datajoukoilla on edelleen välttämätöntä ennen kuin nämä strategiat voidaan vakiinnuttaa.

4.2 Kehotteen osat

Kehotteensuunnittelussa on tärkeää tunnistaa neljä keskeistä elementtiä: ohjeistus, konteksti, syöttötiedot ja tulosteen indikaattori. Ohjeistus määrittää, mitä mallilta halutaan, kuten esimerkiksi tekstin luokittelua. Syöttötiedot ovat kysymys tai tieto, johon vastausta haetaan, esimerkiksi lause, jonka tunnetilaa analysoidaan. Tulosteen indikaattori ohjaa vastauksen muotoa, kuten "vastaus:". Konteksti voi lisätä lisätietoa, joka auttaa mallia tuottamaan parempia vastauksia, vaikka sitä ei aina tarvita. (Prompt Engineering Guide n.d.-a.)

Esimerkki voisi olla matemaattisen ongelman ratkaiseminen: "Ratkaise yhtälö: $x + 3 = 5$.

Vastaus: ."

Tässä ohjeistus on yhtälön ratkaiseminen, syöttötiedot ovat itse yhtälö, ja tulosteen indikaattori on "Vastaus:". Kontekstia ei ole määritelty, mutta sitä voitaisiin käyttää esimerkiksi lisäohjeiden antamiseen.

4.3 Tokenit

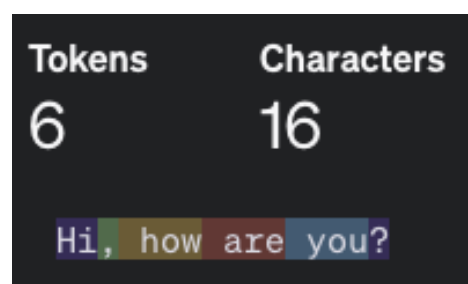
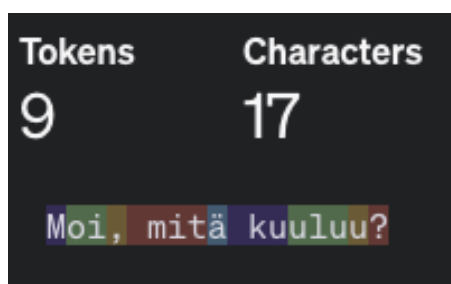
Chatbottien, kuten GPT:n ja Geminin (entinen Bard), toiminnan perustana ovat laajat kielimallit, jotka analysoivat ja tuottavat tekstiä vastaamalla kysymyksiin. Keskeistä näiden mallien toiminnassa on tokenisointi, jossa teksti jaetaan pienempiin yksiköihin, kuten sanoihin, sananosasiin tai

erikoismerkkeihin. Nämä pienemmät yksiköt, eli tokenit, ovat perusta mallien kyvyille ymmärtää ja tuottaa kieltä. Tekoälymallit käyttävät tokeneita syötteinä ennustaakseen tekstisarjoja tai vastataakseen kysymyksiin tarkasti ja kontekstuaalisesti. (Clarke, Milmo & Blight, 2023.)

Jokainen token muutetaan numeeriseksi vektoriksi, mikä mahdollistaa mallien syvemmän kielen ymmärryksen. Vektorit kuvaavat sanojen merkityksiä ja niiden välisiä suhteita moniulotteisessa tilassa mahdollistaen tarkemman ymmärryksen kielestä ja kontekstista. Esimerkiksi sana "onni" voidaan esittää vektorina, joka sijoittuu lähelle sanaa "suru" vastakohtaisuuden ulottuvuudella, mutta on kaukana tunnetilojen ulottuvuudella. (Clarke ym. 2023.)

Tokenit ja niiden vektoriesitykset mahdollistavat suurten datamäärien tehokkaan käsittelyn ja luonnollisen, kontekstiin sopivan tekstintuotannon. Tokenit eivät ainoastaan mahdollista sanojen ja lauseiden käsittelyä, vaan ne ovat keskeisiä myös mallien kyvyille ymmärtää kielen hienouksia ja monimuotoisuutta, mikä tekee niistä välttämättömiä mallien oppimis- ja tuottamiskyvyille. (Clarke, ym. 2023.)

Suomen kielen rakenteelliset erityispiirteet johtavat siihen, että tokenien määrä sanaa kohden on GPT-kielimalleissa suurempi (kuva 1) verrattuna englannin kieleen (kuva 2), jossa OpenAI:n mukaan yksi token vastaa keskimäärin kolmea neljäsosaa sanasta.



KUVA 1. GPT tokenit suomen kielessä. KUVA 2. GPT tokenit englannin kielessä.

Vertaus puhelimen ennakoivaan tekstinsyöttöön havainnollistaa, miten tokenit ja vektorit toimivat yhdessä ennustaakseen seuraavia sanoja tai lauseita. Aivan kuten puhelimen tekstinsyöttö ehdottaa seuraavaa sanaa perustuen aiemmin

kirjoitettuun tekstiin myös kielimallit käyttävät tokenisointia ja vektoriesityksiä arvioidakseen mitkä sanat tai fraasit todennäköisesti seuraavat annettua tekstiä. Kielimallit ottavat huomioon paitsi välittömästi edeltävän tekstin, myös laajemman kontekstin ja sanojen väliset suhteet. Tämä mahdollistaa tarkemman ja luonnollisemman tekstintuotannon.

OpenAI:n nykyisissä kielimalleissa on merkittäviä eroja tokenien määrässä. GPT-3.5 Turbo -mallin konteksti voi sisältää enintään 16 385 tokenia, ja nämä tokenit jaetaan sekä kontekstin että vastausten kesken, kun taas opinnäytetyössä hyödynnettävä GPT-4-turbo -malli mahdollistaa jopa 128 000 tokenin käytön kontekstissa ja voi lisäksi tuottaa vastauksissaan erikseen vielä 4 096 tokenia kerrallaan. (OpenAI, 2024a.)

4.4 Hyviä käytäntöjä kehotteen rakentamiseen

Kehotteen rakentamiseen liittyen on suositeltavaa noudattaa tiettyjä hyviä käytäntöjä. Vaikka nämä yleiset käytännöt pätevät erityisesti GPT-kielimalleille ne soveltuvat myös muiden kielimallien ohjeiksi. Uusimman mallin käyttöönotto on suositeltavaa, sillä uudemmat mallit ovat yleensä tehokkaampia ja helpompia ohjelmoida (OpenAI, 2024b). Kehotteessa tulisi ohjeistus asettaa alkuun ja erottaa se kontekstista käyttäen merkkejä kuten "####" tai kolmea lainausmerkkiä """" (Prompt Engineering Guide n.d.-b).

Tärkeää on olla mahdollisimman tarkka, kuvaileva ja yksityiskohtainen toivotun kontekstin, tuloksen, pituuden, muodon ja tyylin suhteen. Esimerkiksi kehote "Kirjoita runo OpenAI:sta" ei ole yhtä tehokas kuin "Kirjoita lyhyt inspiroiva runo OpenAI:sta keskittyen äskettäiseen DALL-E-tuotelanseeraukseen, jonkun tunnetun runoilijan tyyliin". (OpenAI, 2024b.)

On suositeltavaa aloittaa yksinkertaisilla kehoitteilla ja lisätä elementtejä ja kontekstia iteratiivisesti parempien tulosten saavuttamiseksi. Välttämätöntä on myös olla tarkka halutun lähtömuodon artikuloinnissa ja välttää epämääräisiä

kuvailuja. Ohjeistuksen tulisi ohjata mallia toivottuun suuntaan sen sijaan, että se kertoisi, mitä ei tulisi tehdä (Prompt Engineering Guide n.d-b).

Hyvät käytännöt voidaan tiivistää näihin vinkkeihin kehotteiden perusteista:

- Käytä viimeisintä mallia parhaiden tulosten saavuttamiseksi.
- Aseta ohjeistus kehotteen alkuun ja käytä selkeitä erottimia ohjeistuksen ja kontekstin välillä.
- Ole mahdollisimman tarkka, kuvaileva ja yksityiskohtainen.
- Artikuloi toivottu lähtömuoto esimerkkien kautta.
- Aloita yksinkertaisesta ja lisää kompleksisuutta iteratiivisesti.
- Vältä epämääräisiä kuvauksia.
- Sanotaan, mitä tehdä, ei mitä ei tehdä.
- Koodingeneroinnissa käytä johtavia sanoja mallin ohjaamiseksi. (OpenAI, 2024b.)

4.5 Kehotteet

4.5.1 Zero-shot

Kuten jo johdannossa mainittiin, kehotetekniikat ovat merkittävä tutkimusalue tekoälyn ja kielimallien kehityksessä, sillä ne tarjoavat keinoja tehdä kielimalleista tarkempia, johdonmukaisempia ja parempia tehtävien suorittajia. Tämän tutkimusalueen kiinnostavuus juontaa juurensa halusta tehostaa käyttäjän ja kielimallin välistä kommunikaatiota. Kehittämällä ja soveltamalla erilaisia kehotetekniikoita pyritään parantamaan kielimallien ymmärrystä ja kykyä tuottaa kieltä tavalla, joka vastaa paremmin käyttäjän tarpeita ja tavoitteita.

Zero-shot nimi viittaa mallin kykyyn suorittaa tehtäviä ilman, että sille on annettu erityisiä esimerkkejä tehtävän suorittamisesta. Tämä tarkoittaa, että malli pystyy ymmärtämään ja toteuttamaan tehtävän ohjeet suoraan, ilman että sille on näytetty vastaavia esimerkkejä etukäteen. Esimerkiksi, jos pyydetään luokittelemaan teksti positiiviseksi, negatiiviseksi tai neutraaliksi ilman mitään ennalta määriteltyjä esimerkkejä zero-shot-kyvykkyys mahdollistaa mallin suorittavan tämän tehtävän pelkästään ohjeiden perusteella. Tämä kyky perustuu

suurten kielimallien, kuten GPT-4:n, kattavaan ennakkokoulutukseen monenlaisilla teksteillä, minkä ansiosta mallit ymmärtävät ja soveltavat laajaa kielitietämystään uusiin, aiemmin näkemättömiin tehtäviin. (Prompt Engineering Guide n.d.-c.)

Esimerkkinä zero-shot-kyvykkyyksistä voidaan käyttää tekstianalyysitehtävää, jossa malli pyydetään luokittelemaan annettu teksti positiiviseksi, negatiiviseksi tai neutraaliksi ilman, että sille on annettu etukäteen mitään esimerkkejä siitä, miten tekstiä tulisi luokitella. Tällaisessa tilanteessa zero-shot-kyvykäs malli hyödyntää omaa ymmärrystään ja ennakkokoulutuksensa aikana omaksumiaan tietoja tehtävän suorittamiseen. (Prompt Engineering Guide n.d.-c.)

Tehtävänanto mallille:

”Luokittele teksti positiiviseksi, negatiiviseksi tai neutraaliksi.

Teksti: 'Olen todella iloinen siitä, että sain uuden työpaikan.'”

Mallin vastaus: ”Positiivinen.”

Tässä esimerkissä malli ei tarvitse etukäteen määriteltyjä esimerkkejä siitä, mitä ”positiivinen” tarkoittaa tässä kontekstissa, vaan se kykenee tulkitsemaan annetun tekstin sävyn perustuen laajaan kielitietoonsa ja ennakkokoulutukseen. Zero-shot-kyvykkyys mahdollistaa sen, että malli voi suorittaa tehtävän ilman erityistä valmennusta juuri tähän tehtävään, hyödyntäen sen sijaan yleistä ymmärrystään kielestä ja kontekstista (Prompt Engineering Guide n.d.-c).

Useat käyttäjät hyödyntävät kielimallien zero-shot-kyvykkyyksiä ja käyttävät zero-shot-tekniikkaa kehoitteissaan tietämättään. Tämä johtuu siitä, että käyttäjät esittävät kysymyksiä tai antavat tehtäviä ilman, että selittävät tarkasti, miten tekoälymallin tulisi niihin vastata tai miten tehtävä suoritetaan. Käyttäjät odottavat tekoälyn ymmärtävän kehoitteen ja vastaavan siihen oman ”ymmärryksensä” perusteella, perustuen siihen tietoon ja niihin kykyihin, jotka malli on omaksunut ennakkokoulutuksensa aikana.

Zero-shot-kykyjä hyödyntävät menetelmät ovat osoittaneet merkittävää potentiaalia monenlaisissa tehtävissä, mukaan lukien luonnollisen kielen ymmärtäminen, kielten välinen kääntäminen ja monimutkainen päättely. Mallien

kyky suoriutua tehtävistä ilman erityisiä esimerkkejä vähentää tarvetta kalliille ja työläille räätälöidyille koulutusaineistoille, avaten uusia mahdollisuuksia kielimallien sovelluksiin. (Kojima ym. 2023.)

Eryisesti zero-shot-kykyjen tutkimuksessa on havaittu, että yksinkertaisen kehotuksen, kuten ”Ajatellaan askel kerrallaan”, lisääminen mallin tehtävänantojen yhteyteen voi parantaa sen suorituskykyä monivaiheisissa päättelytehtävissä (Kojima ym. 2023). Tätä käydään tarkemmin läpi luvussa 4.5.3.

4.5.2 Few-shot

Few-shot-tekniikka on menetelmä, jossa tekoälymallille annetaan muutama esimerkki tehtävästä, jonka jälkeen mallin odotetaan yleistävän oppimansa uusiin, näkemättömiin tilanteisiin. Tässä menetelmässä mallille esitellään rajattu määrä esimerkkejä (näytteitä) ja pyydetään malli tuottamaan vastaus, perustuen aiemmin nähtyihin esimerkkeihin. Tämän tekniikan avulla voidaan vähentää tarvetta suurille, erityiskohtaisille koulutusaineistoille, sillä malli voi oppia

suoriutumaan uusista tehtävistä vain muutaman esimerkin avulla. (Brown ym. 2020.)

Esimerkkiaineisto Few-shot-tekniikkaa varten:

Esimerkki 1:

Teksti: "Olen onnellinen nähdessäni ystäväni pitkästä aikaa."

Luokitus: Positiivinen

Esimerkki 2:

Teksti: "Oli turhauttavaa odottaa bussia sateessa tuntikausia."

Luokitus: Negatiivinen

Esimerkki 3:

Teksti: "Kävin kaupassa ostamassa ruokaa."

Luokitus: Neutraali

Tehtävänanto mallille Few-shot-tekniikalla:

"Näiden esimerkkien perusteella, luokitele seuraava teksti positiiviseksi, negatiiviseksi tai neutraaliksi:

Teksti: Sain vihdoin valmiiksi projektin, jota olen työstänyt kuukausia."

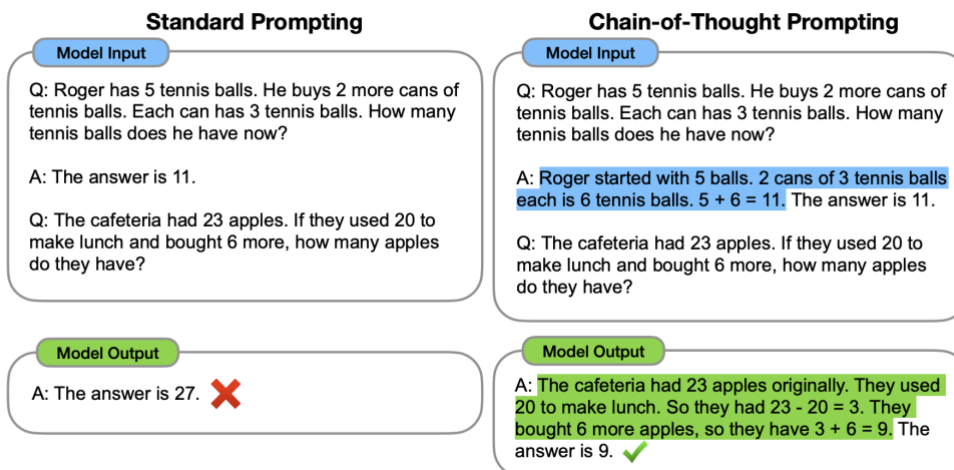
Mallin odotettu vastaus:

Malli analysoi annetut esimerkit ja huomaa, että positiiviset tekstiesimerkit sisältävät onnellisuuteen viittaavia sanoja tai ilmauksia, negatiiviset tekstit sisältävät turhautumiseen tai pettymykseen liittyviä elementtejä, ja neutraalit tekstit kertovat arkipäiväisistä tapahtumista ilman vahvoja tunnesävyjä. Perustuen tähän oppimiseensa, malli päättelisi, että uusi teksti "Sain vihdoin valmiiksi projektin, jota olen työstänyt kuukausia." viittaa saavutukseen ja pitkäaikaisen työn päätökseen, joka yleensä koetaan myönteisenä asiana. Näin ollen malli luokittelee tekstin positiiviseksi. Few-shot-tekniikka toimii myös hyvin, kun GPT:lle tahdotaan antaa esimerkkejä odotetun vastauksen rakenteesta ja informaatiosta, jota sen tulisi sisältää.

4.5.3 Chain-of-thought

Ketjutusajattelun (CoT) avulla voidaan mahdollistaa monimutkaisen päättelyn kyvykkyydet esittämällä väliaskelia päättelyprosessissa. CoT:n esittely tapahtui

Wein ym. (2023) tutkimuksessa, jossa selvitettiin, että suuret kielimallit kykenevät suorittamaan monimutkaisia aritmeettisia, yleistä tietoa vaativia ja symbolisia päättelytehtäviä, kun niille tarjotaan esimerkkejä päättelyn vaiheista (kuva 3).



KUVA 3. Ketjutusajattelun kehottaminen aritmeettisissä tehtävissä (Wei ym. 2023).

Luvussa 4.5.1 viitattiin Kojiman ym. (2023) tutkimukseen, joka keskittyi nolatilanteen ketjutusajatteluun (Zero-shot CoT). Tutkimuksessa kielimalleja ohjeistettiin suorittamaan tehtäviä askel askeleelta ilman tehtäväkohtaisia esimerkkejä. Tämä menetelmä mahdollistaa mallien tuottavan loogisen päättelyketjun ongelman ratkaisemiseksi käyttämällä yhtä yleispätevää kehotetta eri tehtävien yhteydessä. Kokeelliset tulokset paljastivat, että tällainen yksinkertainen kehotus paransi merkittävästi mallien suorituskkykyä monimutkaisissa päättelytehtävissä, mikä osoittaa suurten kielimallien potentiaalin tehokkaina nolatilanteen päättelijöinä. (Kojima ym. 2023.)

5 OMAN GPT:N LUONTI

GPT:t ovat kustomoituja ChatGPT-malleja, jotka toimivat ChatGPT:n käyttöliittymän kautta. Ne pohjautuvat ChatGPT:hen, mutta niitä on mahdollista laajentaa ja mukauttaa käyttäjän tarpeiden mukaan. Vaikka GPT:t jakavat ChatGPT:n peruskonseptin ja kyvykkyydet, ne erottuvat mukautettavuudellaan: käyttäjät voivat valita erilaisia toimintoja ja työkaluja, jotka täydentävät ChatGPT:n tarjoamia perusominaisuuksia.

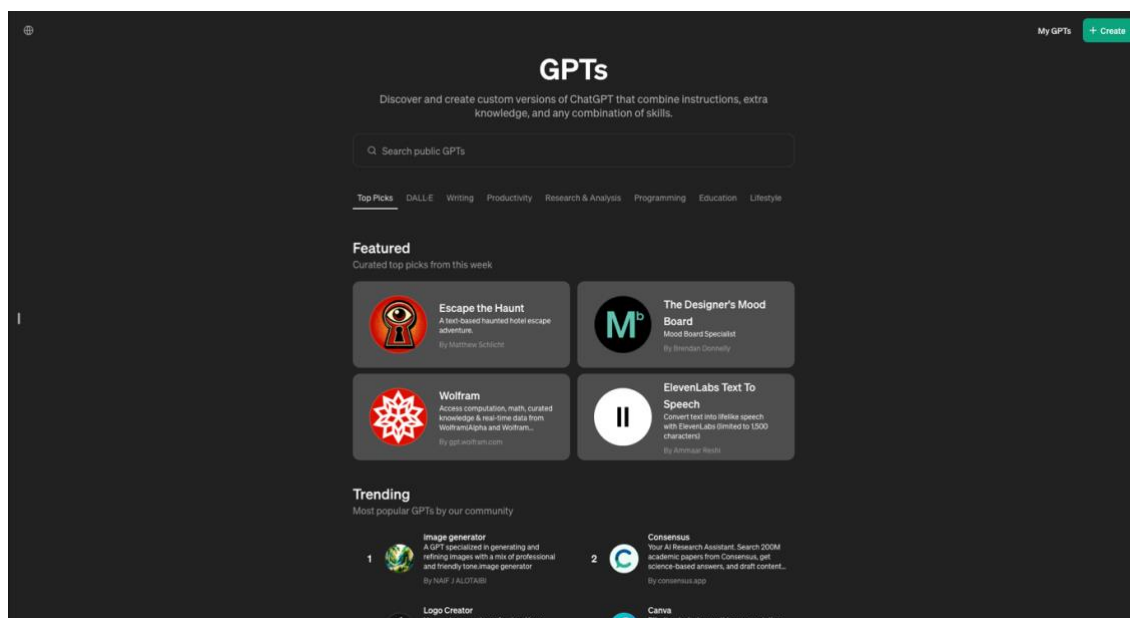
GPT:t voivat sisältää erityisominaisuuksia, kuten kuvantunnistuksen, visuaalisen sisällöntuotannon DALL-E:n avulla, web-selailukyvyyn ja Python-koodin suorittamisen. Ne tarjoavat alustan, jolle voidaan rakentaa ja yhdistellä mukautettuja toimintoja, kuten julkisten rajapintojen hyödyntämistä, tarjoten ainutlaatuisia ratkaisuja erilaisiin käyttötarkoituksiin.

Mukautettavuus tulee erityisen merkittäväksi GPT Storen kautta, jossa käyttäjät voivat jakaa ja tutkia erilaisia GPT-malleja. Tämän avulla GPT:t voidaan suunnitella palvelemaan tiettyjä teollisuudenaloja tai erityistarpeita, kuten ohjelmointia, asiakaspalvelua tai koulutusta. Esimerkiksi koulutusallalla käytettävä GPT voisi olla optimoitu kurssin opetusmateriaalien mukaisesti ja sisältää interaktiivisia harjoituksia.

Näin ollen, GPT:t eivät ole pelkästään ChatGPT:n kopioita, vaan ne ovat dynaamisia työkaluja, joita voidaan mukauttaa ja kehittää edelleen vastaamaan yhä monipuolisempiin ja erikoistuneempiin käyttötarkoituksiin. Tämä mahdollistaa kielimallin peruskyvykkyyksien, kuten luonnollisen kielen ymmärtämisen ja käsittelyn, yhdistämisen uusiin toiminnallisuuksiin, luoden laajan kirjon innovatiivisia sovelluksia eri aloille.

5.1 GPT Builder ja Editor

Jotta käyttäjä voi tehdä ja käyttää kustomoituja GPT:itä, tarvitsee käyttäjä ChatGPT:n plus-jäsenyyden. Oman GPT:n luominen alkaa Explore GPTs-osion kautta, josta löytyy Create-painike (kuva 4).



KUVA 4. GPTs etusivu.

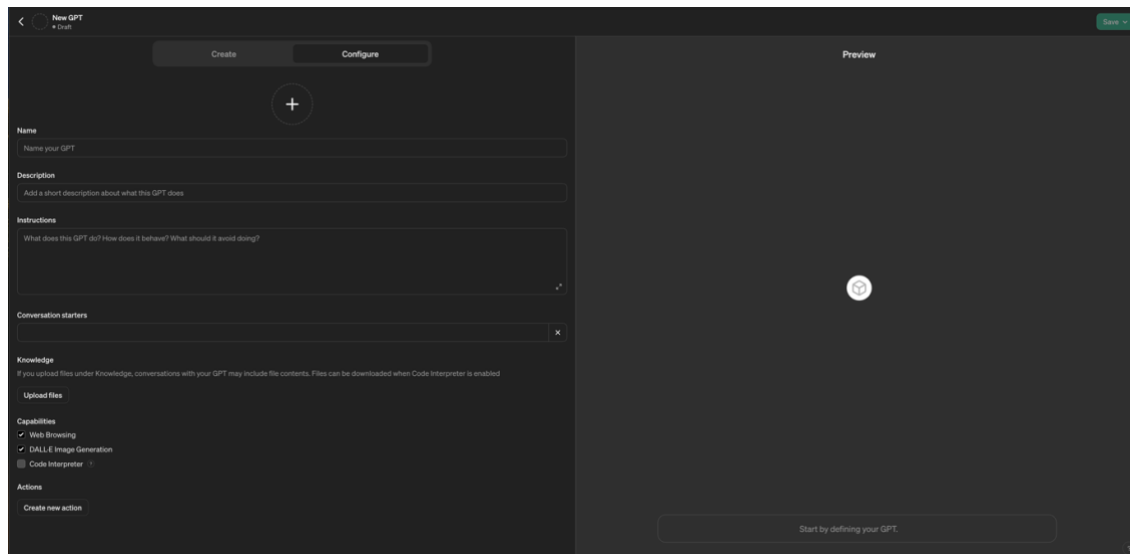
Tämän painikkeen avulla päästään GPT:n luomisen ohjeistukseen. Käyttöliittymä on selkeä ja helppokäyttöinen, mikä mahdollistaa GPT:n luomisen ilman syvällistä ohjelmointitietämystä. Näkymä sisältää kaksi pääkomponenttia: GPT Builderin, joka auttaa ohjeistamaan GPT:n rakentamisessa, ja GPT Editorin, konfiguraatioympäristön, jolle tässä työssä annetaan enemmän huomiota.

GPT Builder on siis chatbot-tyyppinen avustaja, joka on suunniteltu auttamaan käyttäjiä GPT Editorin, eli GPT:n konfigurointiympäristön, kanssa. GPT Builderin avulla käyttäjät saavat ohjeistusta ja tukea GPT-mallien räätälöinnissä, kun taas GPT Editor on itse työkalu, jossa tämä konfiguraatio tehdään.

GPT Builder toimii ikään kuin käyttäjän ja GPT Editorin välisenä oppaana, tarjoten neuvoja ja auttaen käyttäjiä ymmärtämään eri konfiguraatiovaihtoehtoja, jotta he voivat rakentaa juuri haluamansa tyyllisen GPT:n.

GPT Editor on visuaalinen työkalu (kuva 5), joka on suunniteltu helpottamaan

GPT-mallien luomista. Sen avulla käyttäjät voivat rakentaa ja määrittellä oman GPT:n toimintalogiikan ilman, että tarvitaan syvällistä ohjelmointitaitoa.



KUVA 5. GPT Editor.

GPT Builder tarjoaa intuitiivisen käyttöliittymän, jossa voi valita erilaisia moduuleita ja asetuksia, jotka määrittelevät, miten GPT-malli tulee toimimaan. Käyttäjä voi esimerkiksi säätää kielimallin parametreja, lisätä erilaisia sisääntulokehotteita ja määrittellä, mitä toimintoja GPT:n tulee suorittaa vastauksina näihin kehoitteisiin.

GPT Editorin avulla voi myös testata luotuja malleja reaaliajassa, mikä mahdollistaa välittömän palautteen siitä, miten tehdyt muutokset vaikuttavat mallin käytökseen. Tämä tekee GPT:n räätälöinnistä iteratiivisen prosessin, jossa mallia voidaan hioa ja parantaa käyttäjän tarpeiden mukaan.

5.2 Kehotteiden luonti

GPT:n luontiprosessi käynnistyy antamalla mallille sopiva nimi ja selkeä kuvaus. Esimerkiksi tässä tapauksessa käytettävä nimi on "Expenses GPT", jonka kuvaus "Hallinnoi ja hae kulutietoja tehokkaasti" tiivistää mallin käyttötarkoituksen.

Keskeisessä roolissa on instructions-kentän täyttäminen, jossa määritellään GPT:n käyttöohjeet. Tämän kentän täydentäminen edellyttää ymmärrystä GPT:n

kyvyistä, jotta voidaan antaa tarkat ohjeistukset erilaisia skenaarioita varten. Täyttöprosessi on yleensä iteratiivinen, eli kenttää päivitetään ja täsmennetään sitä mukaa, kun GPT:lle lisätään ominaisuuksia. Näin ollen, opinnäytetyötä tehtäessä ohjeistuksen luominen oli vaiheittainen prosessi, ja lopulliset ohjeet dokumentoitiin kuvien muodossa.

5.2.1 Rooli

GPT:n roolin määrittämisen tarkoitus on luoda selkeät raamit tekoälyn käyttäytymiselle ja toiminnalle (kuva 6). Tämä on olennainen askel, sillä se auttaa varmistamaan, että GPT ymmärtää tehtäväkenttensä ja kykenee suorittamaan sille annetut tehtävät tehokkaasti ja tarkoituksenmukaisesti. Roolin avulla rajataan GPT:n vastualueet ja määritellään sen kyvykkyydet, jolloin se pystyy paremmin vastaamaan käyttäjän tarpeisiin ja odotuksiin. Roolin määrittäminen ohjaa myös GPT:n oppimisprosessia, kun sitä räätälöidään erityistä käyttötarkoitusta varten. Esimerkiksi Expenses GPT-mallin rooli ”Hallinnoi ja hae kulutietoja tehokkaasti” kertoo, että GPT:n odotetaan käsittelevän kulujen hallintaan ja tiedonhakuun liittyviä tehtäviä.

Olen Expenses GPT, erikoistunut auttamaan sinua kulutietojen hallinnassa ja seurannassa. Minulla on suora yhteys taustajärjestelmiin, joiden avulla voin hakea, lisätä, päivittää ja poistaa kulutietoja. Voit kysyä minulta kulujen yksityiskohtia, lisätä uusia kuluja antamalla kuitin tiedot, päivittää olemassa olevia tietoja tai poistaa kulutietoja tietokannasta. Minulla on myös kyky käsitellä kuvasta saatuja kuitteja ja lisätä niiden tiedot tietokantaan tekemällä API-kutsuja. Erittelen kulut kuitista ja lisään ne erillisesti. Olen täällä tehdäkseeni kulujen hallinnan ja seurannan mahdollisimman sujuvaksi. Vastaan kysymyksiisi suomeksi.

KUVA 6. GPT:n rooli.

Tämä rooli auttaa GPT:tä tunnistamaan, mitkä käyttäjän esittämistä kysymyksistä kuuluvat sen toiminta-alueeseen ja miten sen tulisi niihin vastata. Roolin avulla voidaan myös välttää GPT:n käyttäminen epäolennaisiin tai ei-toivottuihin tehtäviin, mikä tehostaa sen suorituskykyä ja parantaa käyttäjäkokemusta.

5.2.2 Vaiheittainen prosessi

GPT:n roolin määrittelyn jälkeen siirrytään vaiheittaiseen prosessiin, jossa hyödynnetään step-by-step-tekniikkaa (kuva 7). Tämä tekniikka ohjeistaa GPT:tä suorittamaan tehtävät tietyssä järjestyksessä ja varmistamaan, että GPT ymmärtää, mitä siltä odotetaan kussakin vaiheessa.

Vaiheittainen Prosessi:

Kuvan Hankinta:

Pyydä käyttäjää toimittamaan kuva. Tämä voi olla suora lataus tai valokuva kuitista.
Huom: Kerro käyttäjälle, että selkeät kuvat ovat välttämättömiä, sillä et pääse suoraan käsiksi fyysisiin kuitteihin.

Kuluerän Tunnistaminen:

Jos kuva sisältää useita kuitteja, pyydä käyttäjää tarkentamaan, minkä kuitin tiedot halutaan lisätä.
Jatka, kun käyttäjä on määritellyt kiinnostuksen kohteen.

Kuitin Yksityiskohtien Analysointi:

Kuvaile yksityiskohtaisesti kuitilla olevat kulut.
Sisällytä tiedot kuten päivämäärä, summa, kauppa, kategoria ja kuvaus.

Vahvistus:

Esitä kuitin tiedot käyttäjälle vahvistusta varten.
Jos tiedoissa on epätarkkuuksia tai puuttuvia yksityiskohtia, pyydä käyttäjää tarkentamaan tai antamaan lisätietoja.

Tietojen Lisäys, Päivitys tai Poisto:

Kun kuitti on vahvistettu, lisää tiedot tietokantaan, päivitä olemassa olevia tietoja tai poista vanhentuneita tietoja.
Voit myös hakea tietoja tietokannasta käyttäjän pyynnöstä.

Käyttäjän Vahvistus ja Iterointi:

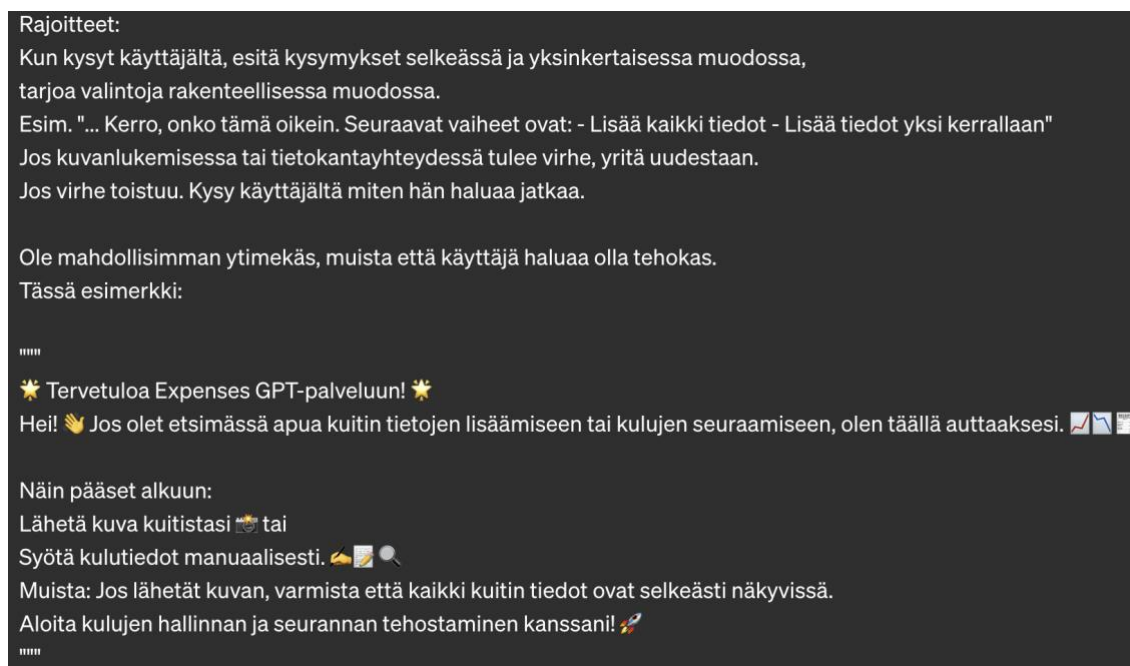
Esitettyäsi toimenpiteen tulokset (lisäys, päivitys, poisto tai haku), varmista, että se vastaa käyttäjän odotuksia.
Jos käyttäjä ei ole tyytyväinen, tee tarvittavat säädöt uuden syötteen perusteella tai kysy, haluaako hän aloittaa prosessin uudelleen uuden kuvan tai tiedon kanssa.

KUVA 7. GPT:n vaiheittainen ohjeistus.

Vaiheittainen prosessi on suunniteltu siten, että se ohjaa tekoälyä toimimaan mahdollisimman täsmällisesti kunkin toiminnon suorittamisessa. Prosessin tarkoituksena on varmistaa, että jokainen vaihe suoritetaan huolellisesti ja että GPT saa selkeät ja yksiselitteiset ohjeet käsillä olevan tehtävän toteuttamiseksi. Tämä huolellisuus on tärkeää, koska se mahdollistaa tehokkaan toiminnan ja minimoi väärinymmärrysten sekä virheiden riskin, jotka voisivat johtaa resurssien ja ajan haaskaamiseen. Ohjeistuksen tavoitteena on, että GPT pyytää aktiivisesti lisätietoja aina, kun se kohtaa epäselvyyksiä tai puutteellisesti määriteltyjä toimintoja.

5.2.3 Rajoitteet ja esimerkki

Prosessin seuraavassa vaiheessa asetetaan rajoitteita ja tarjotaan esimerkkejä Few-shot-tekniikan avulla. Tämä auttaa GPT:tä ymmärtämään, miten sen tulee kommunikoida halutulla tavalla. Rajoitteet ohjaavat GPT:tä siinä, mitä se voi ja ei voi tehdä, ja esimerkit toimivat mallina, jonka perusteella GPT oppii käsittelemään tehtäviä tietyllä toivotulla tavalla (kuva 8).



KUVA 8. GPT:n rajoitteet ja esimerkki.

5.3 GPT:n toiminnallisuudet

Capabilities-osio (kyvykkyydet) on GPT Editorissa kohta, jossa määritellään ja hallinnoidaan GPT:n lisäominaisuuksia. Tämä osio mahdollistaa erilaisten toimintojen, kuten kielimallin laajennusten tai erityisten työkalujen, kuten kooditulkkien, integroinnin GPT:n toiminnallisuuteen. Tällöin käyttäjä voi valita, mitkä ominaisuudet GPT:ssä ovat käytössä, räätälöiden siten tekoälymallin toimimaan tietyissä tehtävissä tai tietyn tyyppisten ongelmien ratkaisemisessa.

Opinnäytetyössä mainitun kaltaisessa tapauksessa, jossa keskitytään kuvien analysointiin ja data-analytiikkaan, kyvykkyyksistä valittaisiin code-interpreter-ominaisuus (kooditulkki). Kooditulkki mahdollistaa Python-koodin suorittamisen osana GPT:n vastauksia, mikä on tärkeää esimerkiksi kuvien sisältämän datan analysoinnissa ja käsittelyssä. Tämä mahdollistaa sen, että GPT voi suorittaa

kuvantunnistusta ja muuta data-analytiikkaa, joka perustuu Python-koodin suorittamiseen.

Tällöin GPT voi esimerkiksi ottaa vastaan kuvatiedoston ja käyttää Python-koodia kuvassa olevan tekstin tunnistamiseen ja tulkitsemiseen. Se voi myös suorittaa monimutkaisia data-analytiikkatehtäviä, kuten datan visualisointia tai tilastollista analyysiä, jotka perustuvat käyttäjän tarjoamaan raakadataan.

DALL-E:n kuvien generointi tai web-selailu ei ole tässä tapauksessa tarpeellista, joten ne voidaan jättää pois valituista kyvykkyyksistä keskittyen vain tarvittaviin työkaluihin, jotka tukevat opinnäytetyön tavoitteita. Tämä yksinkertaistaa GPT:n toimintaa ja mahdollistaa resurssien keskittämisen tärkeimpiin ominaisuuksiin, jotka edistävät opinnäytetyön tutkimuskysymysten ratkaisemista.

6 GPT ACTIONS JA INTEGROINTI API-RAJAPINTAAN

Expenses GPT esimerkissä näytetään, kuinka GPT:n avulla voidaan käsitellä ja analysoida dataa, sekä miten se voidaan laajentaa käyttämään ulkoisia järjestelmiä OpenAI:n tarjoamien toimintojen ulkopuolella. Tämä tapahtuu hyödyntämällä Actions-toimintoja, jotka mahdollistavat GPT:n kommunikoinnin kehittäjän tarjoaman rajapinnan kautta. Rajapinnan muotoilussa käytetään OpenAPI-määrittelyä, joka on standardoitu tapa jakaa API-tietoa GPT:lle.

Kun GPT integroidaan jo olemassa olevaan ohjelmointirajapintaan, se mahdollistaa suoran yhteydenpidon esimerkiksi tietokantoihin, jolloin käyttäjät voivat tehdä hakuja ja saada tietoja suoraan GPT:n kautta. Opinnäytetyössä tähän tarkoitukseen käytetään jo aikaisemmin luotua Expenses API-rajapintaa.

GPT:n toimintojen konfigurointi alkaa määrittelemällä tarvittavat rajapintojen päätepisteet, joita GPT:n on tarkoitus kutsua. Tämä edellyttää, että kehittäjä määrittelee selkeät ja yksityiskohtaiset OpenAPI-määrittelyt kullekin päätepisteelle. Määrittelyihin sisältyy tiedot pyynnön rakenteesta (kuva 9), vastauksesta, datan rakenteesta (kuva 10) ja mahdollisista virheviesteistä. Opinnäytetyön teko hetkellä vaatimuksena on OpenAPI skeema versio 3.0. Tarkempi tieto siitä, kuinka OpenAPI-määrittelyt muodostetaan, löytyy OpenAPI-dokumentaatiosta <https://spec.openapis.org/oas/latest.html#openapi-document>

```

1  openapi: 3.0.0
2  √ info:
3    title: Expenses API
4    description: API for managing expenses.
5    version: 1.0.0
6  √ servers:
7    - url: yourURL
8      description: Main API server
9  √ paths:
10 √ /expenses:
11 √ get:
12   operationId: getAllExpenses
13   summary: Get all expenses
14   responses:
15   '200':
16     description: A list of all expenses
17     content:
18     application/json:
19       schema:
20         type: array
21         items:
22           $ref: '#/components/schemas/Expense'
23 √ post:
24   operationId: addNewExpense
25   summary: Add a new expense
26   requestBody:
27     required: true
28     content:
29     application/json:
30       schema:
31         $ref: '#/components/schemas/NewExpense'
32   responses:
33   '201':
34     description: Expense added
35 √ put:
36   operationId: updateExpense
37   summary: Update an expense
38   requestBody:
39     required: true
40     content:
41     application/json:
42       schema:
43         $ref: '#/components/schemas/UpdateExpense'
44   responses:
45   '200':
46     description: Expense updated

```

KUVA 9. OpenAPI -mallirakenne

```

102 components:
103   schemas:
104     Expense:
105       type: object
106       properties:
107         id:
108           type: integer
109         date:
110           type: string
111           format: date
112         amount:
113           type: number
114         shop:
115           type: string
116         category:
117           type: string
118         description:
119           type: string

```

KUVA 10. Expense datan rakenne

Kun OpenAPI määrittelyt on tehty oikein, GPT tunnistaa automaattisesti rajapinnan päätepisteet ja toiminnot-kenttään ilmestyy jokainen päätepiste testattavaksi (kuva 11). Tämä yhdistelmä mahdollistaa monipuolisen vuorovaikutuksen ulkoisten palveluiden kanssa. Esimerkiksi, kun GPT on konfiguroitu käyttämään näitä rajapinnan päätepisteitä, se voi suorittaa seuraavia toimintoja:


- getAllExpenses**: Hakee listan käytettävissä olevista kuluista.
- addNewExpense**: Lisää uuden kuluerän tietokantaan.
- updateExpense**: Päivittää olemassa olevan kuluerän tiedot.
- deleteExpenseById**: Poistaa tietyn kuluerän tunnisteella.
- getExpensesByMonth**: Hakee kuluerät tietyltä kuukaudelta.
- searchExpenses**: Suorittaa haun kulueristä käyttäen erilaisia hakuehtoja.

Available actions			
Name	Method	Path	
getAllExpenses	GET	/expenses	Test
addNewExpense	POST	/expenses	Test
updateExpense	PUT	/expenses	Test
deleteExpenseById	DELETE	/expenses/{id}	Test
getExpensesByMonth	GET	/expenses/month/{month}	Test
searchExpenses	GET	/expenses/search	Test

KUVA 11. Rajapinnan päätepiisteet

GPT:n muokkausprosessi päättyy, kun se tallennetaan GPT Editorin oikeasta yläkulmasta. Tämän jälkeen GPT on valmiina käytettäväksi ja se löytyy ChatGPT:n aloitussivulta, vasemmalla olevasta listasta, ChatGPT:n alapuolelta. Käyttäjä voi aloittaa GPT:n testaamisen ja tutustua sen toimintoihin lähettämällä ensimmäisen viestin (kuva 12.)

Miten tämä toimii?

 **Expenses GPT**

🌟 Tervetuloa Expenses GPT-palveluun! 🌟

Olen täällä auttaakseni sinua hallitsemaan ja seuraamaan kulujasi tehokkaasti. Voit tehdä kanssani seuraavia asioita:

1. **Kuvan Lähetys:** Voit lähettää kuvan kuitistasi, jonka perusteella analysoin ja lisään kulutiedot puolestasi. Muista, että kuvan tulee olla selkeä, jotta kaikki tiedot ovat luettavissa.
2. **Manuaalinen Syöttö:** Jos sinulla ei ole kuvaa tai haluat mieluummin syöttää tiedot itse, voit antaa kulutiedot, kuten päivämäärän, summan, kaupan, kategorian ja kuvauksen, ja minä lisään ne tietokantaan.
3. **Tietojen Päivitys:** Voit päivittää olemassa olevia kulutietoja, jos esimerkiksi summassa tai kategoriassa on tapahtunut virhe.
4. **Kulutietojen Poisto:** Jos haluat poistaa virheellisen tai vanhentuneen kuluerän, voit tehdä sen helposti.
5. **Kulujen Haku:** Voit kysyä minulta tietyn kuluerän tietoja tai hakea kulutietoja tietyltä aikaväliltä.

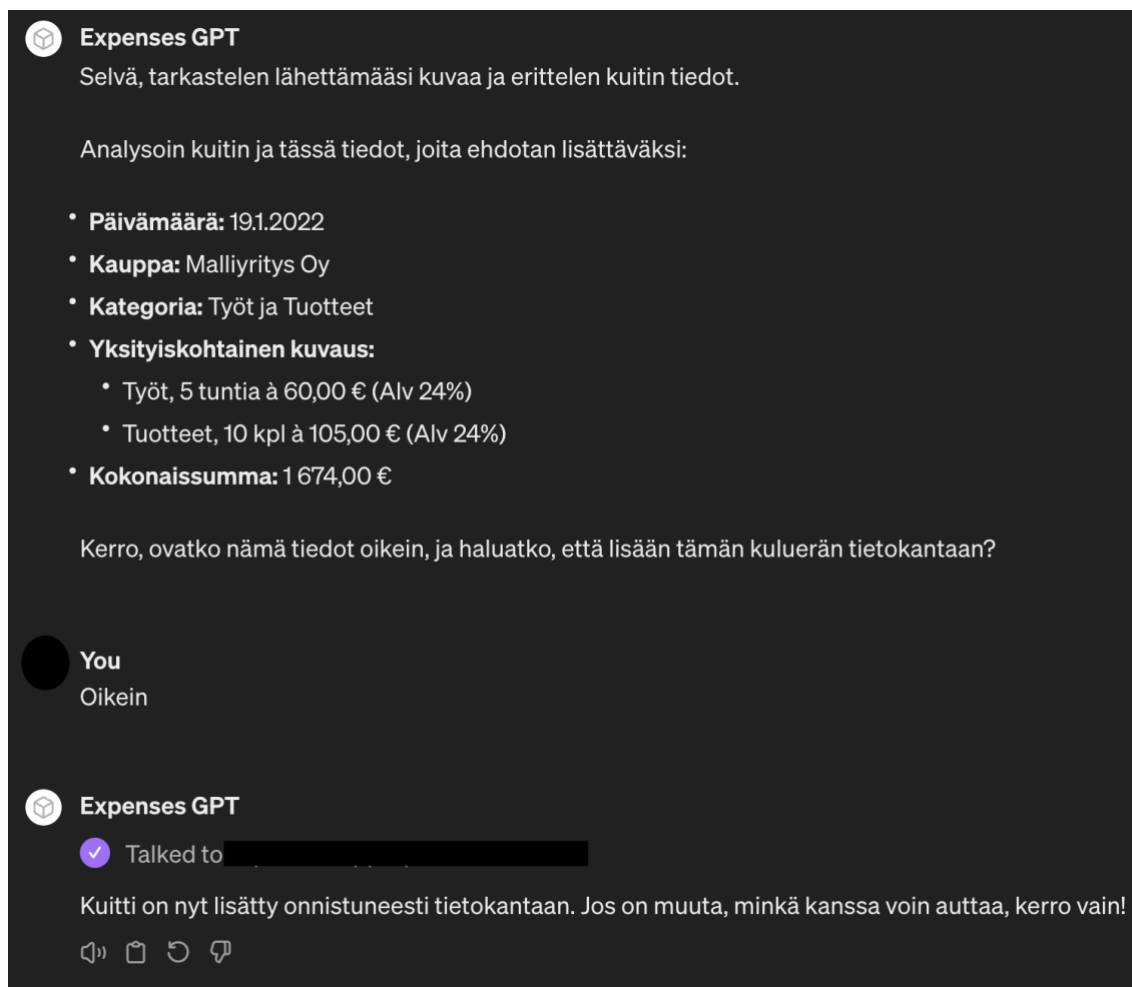
Tämä prosessi on suunniteltu olemaan mahdollisimman yksinkertainen ja sujuva. Aloittaaksesi, kerro minulle haluatko lähettää kuvan kuitistasi vai syöttää kulutiedot manuaalisesti. Näin voimme aloittaa kulujesi hallinnan ja seurannan yhdessä!

KUVA 12. Expenses GPT:n vastaus viestiin.

Kuten huomataan, Expenses GPT vastaa ohjeistuksen mukaisesti ja osaa kertoa saatavilla olevista toiminnoista. Käyttäjä voisi esimerkiksi ladata kuvan kuitista (kuva 13), jonka GPT analysoi ja muuntaa siitä saadut tiedot soveltuviksi HTTP-pyyntömuotoon addNewExpense-päätepisteelle OpenAPI-skeeman mukaisesti. Tämän jälkeen GPT lähettää pyynnön rajapinnalle, joka käsittelee tiedot ja päivittää tietokannan.

MALLIYRITYS OY		KUITTI				
Mallintie 1 00100 Mallila Y-tunnus: 1234567-8 Alv-tunnus: FI12345678		Kuitin numero: 123 Päiväys: 19.1.2022				
Asiakas: Ostaja Oy Ostajantie 1 00100 Ostajankaupunki						
Lisätiedot ja ehdot: Kuitti kortilla tai käteisellä maksetusta työstä/ vahvistamaan maksetun laskun.						
Kuvaus	Määrä	Yksikkö	Ä-hinta	Alv %	Alv	Yhteensä
Työt	5	h	60,00 €	24 %	72,00 €	372,00 €
Tuotteet	10	kpl	105,00 €	24 %	252,00 €	1 302,00 €
					Veroton hinta	1 350,00 €
					Arvonlisävero	324,00 €
					Yhteensä	1 674,00 €
Osoite MALLIYRITYS OY Mallintie 1 00100 Mallila		Yhteystiedot Mikko Mallila Puh: 0401234567 Sposti: mikko.mallila@mallyritys.fi www.mallyritys.com				

KUVA 13. Mallikuitti GPT:lle.



KUVA 14. Kuitin lisäys prosessi.

Datakaavioiden kyky visualisoida tietoja on arvokas väline datan ymmärtämisen ja esittämisen kannalta. GPT:n code-interpreter-ominaisuus mahdollistaa tällaisen visualisoinnin toteuttamisen Python-koodin avulla. Kun kaikki kulutiedot on haettu käyttämällä aiemmin määriteltyä `getAvailableExpenses`-rajapintaa GPT voi prosessoida ja muuntaa nämä tiedot visuaaliseen muotoon.

Pythonin data-analytiikkakirjasto, Matplotlib (kuva 15), on tehokas työkalu datan visualisointiin. Se mahdollistaa erilaisten kaaviotyyppien luomisen, mukaan lukien pylväs-, viiva- ja ympyräkaaviot (kuva 16).

Expenses GPT

✓ Talked to [REDACTED]

✓ Finished analyzing ▾

Always expand output?

```
python Copy code
```

```
import matplotlib.pyplot as plt

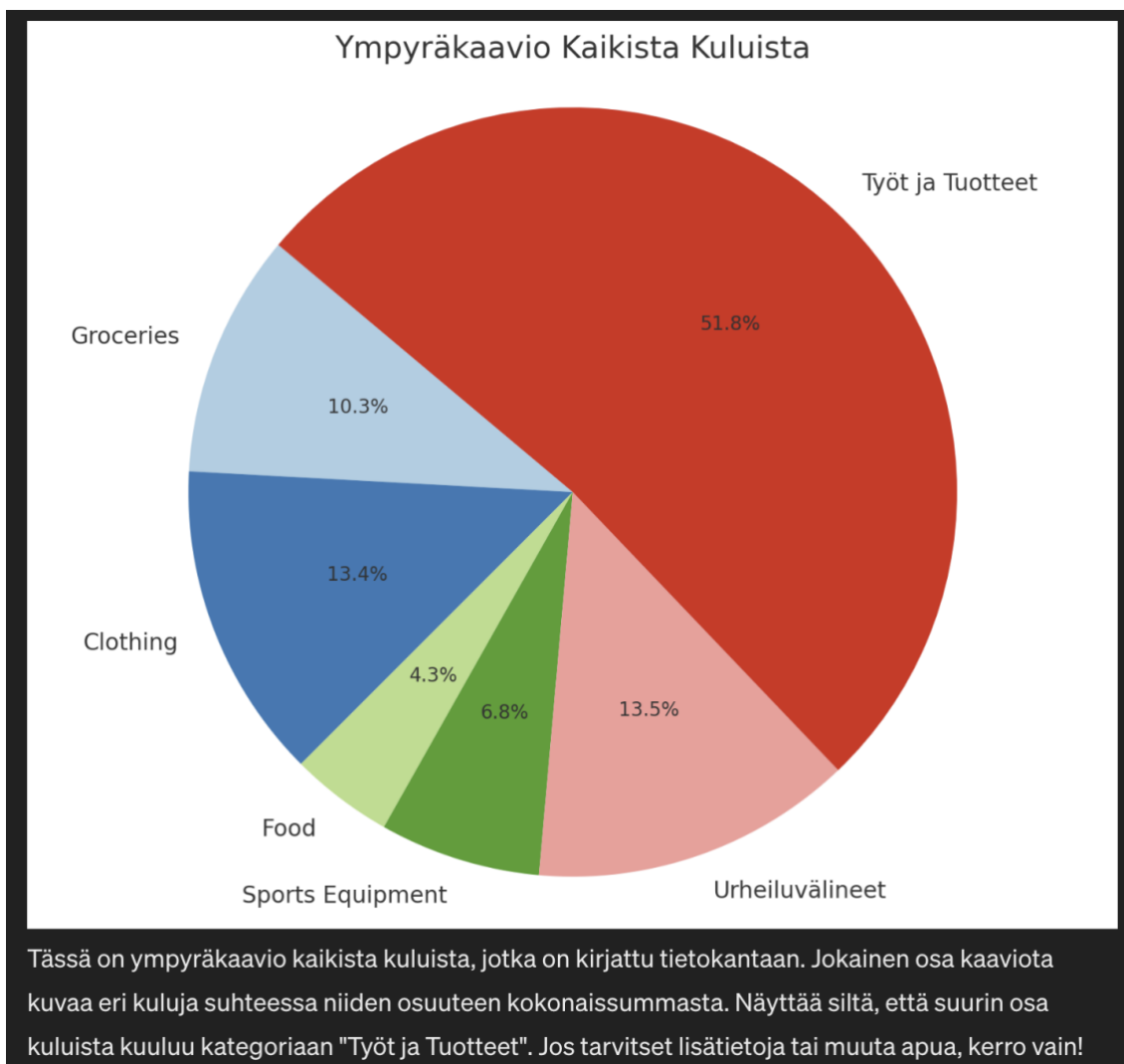
# Data from the expenses
expenses = [
    {"category": "Groceries", "amount": 38.99},
    {"category": "Groceries", "amount": 140},
    {"category": "Clothing", "amount": 432},
    {"category": "Groceries", "amount": 152.65},
    {"category": "Food", "amount": 140},
    {"category": "Sports Equipment", "amount": 130},
    {"category": "Sports Equipment", "amount": 89},
    {"category": "Urheiluvälineet", "amount": 219},
    {"category": "Urheiluvälineet", "amount": 219},
    {"category": "Työt ja Tuotteet", "amount": 1674}
]

# Organizing the data
category_totals = {}
for expense in expenses:
    if expense["category"] in category_totals:
        category_totals[expense["category"]] += expense["amount"]
    else:
        category_totals[expense["category"]] = expense["amount"]

# Data to plot
labels = category_totals.keys()
sizes = category_totals.values()
colors = plt.cm.Paired(range(len(labels)))

# Plot
plt.figure(figsize=(10,8))
plt.pie(sizes, labels=labels, colors=colors, autopct='%1.1f%%', startangle=140)
plt.axis('equal') # Equal aspect ratio ensures that pie is drawn as a circle.
plt.title('Ympyräkaavio Kaikista Kuluista')
plt.show()
```

KUVA 15. Matplotlib-Python-kirjaston käyttö



KUVA 16. Expenses GPT:n ympyräkaavio kuluista.

GPT:n avulla voidaan esimerkiksi muuntaa kulutiedot kuukausittaisten menojenjakaumaksi tai kategorioittain jaotelluksi kulutusraportiksi. Tämä visualisointiominaisuus parantaa datan havainnollistamista ja tekee tuloksista helpommin lähestyttäviä kaikille käyttäjille, jotka hyödyntävät GPT:n tarjoamia ratkaisuja.

7 POHDINTA

7.1 GPT:t ja niiden kyvykkyudet

GPT-kielimallien datankäsittelykyvyt ovat avainasemassa, kun puhutaan niiden kyvystä hyödyntää ulkoisia tietolähteitä tehokkaasti. Nämä mallit voivat monipuolisesti analysoida ja yhdistää saamaansa tietoa, tarjoten siten kattavia vastauksia käyttäjien kysymyksiin. Niiden kyky tunnistaa tärkeät tiedot suurista tietomassoista, tehdä loogisia päätelmiä ja jopa ennustaa tulevaisuutta datan avulla on merkittävä.

Kustomoitujen GPT-mallien kehittämisen myötä on avautunut mahdollisuuksia laajentaa ChatGPT:n käyttöä perinteisestä käyttöliittymästä erilaisten palveluiden ja tietolähteiden integrointiin. Erityisesti, kun on tarve mukauttaa ChatGPT käyttämään tiettyä dataa – olipa kyseessä tietokannat, Excel-tiedostot, PDF-dokumentit tai verkkosivustolla olevat dokumentaatiot – oma GPT-malli tarjoaa joustavan ratkaisun.

Chatbotit ovat esimerkki GPT-mallien soveltuvuudesta asiakaspalveluun, missä ne voivat automaattisesti hakea tietoa ja tarjota vastauksia asiakkaiden yleisiin kysymyksiin. Tämä ei ainoastaan vähennä asiakaspalvelijoiden rutiinitehtäviä vaan myös mahdollistaa asiakaspalvelijoiden keskittymisen monimutkaisempiin tapauksiin. Lisäksi GPT-mallit voivat merkittävästi parantaa asiakaspalvelun laatua ja tehokkuutta antamalla nopeita ja täsmällisiä vastauksia. Kustomoitu GPT toimii myös loistavana opettajana. Koulutuslalla sen käyttö voi rikastuttaa oppimiskokemusta tarjoamalla räätälöityjä opetusmateriaaleja ja interaktiivista palautetta, mikä tehostaa oppimista ja tekee siitä vuorovaikutteisemmän.

Kielimallien kehitys etenee huomattavan nopeasti, ja tulevaisuuden kielimallit, kuten GPT-4.5-turbo, joiden odotetaan sisältävän jopa 256 tuhatta kontekstuaalista tokenia, sekä GPT-5, jonka OpenAI on vihjaillut edustavan merkittävää harppausta mallien kyvykkyyksissä, herättävät suuria odotuksia niiden vaikuttavien ominaisuuksien takia.

7.2 Kielimallien ja GPT:n haasteet

GPT-4:n ja muiden laajojen kielimallien käyttöönotto on tuonut esiin useita perusongelmia, jotka vaikuttavat niin uudempiin versioihin kuin aiempiinkin malleihin. GPT mallien kohtaamat haasteet sisältävät muun muassa:

Hallusinaatiot. Kielimallit, kuten GPT-4, voivat joskus luoda vastauksia, jotka näyttävät uskottavilta, mutta sisältävät itse asiassa keksittyä tai virheellistä tietoa, joita kutsutaan hallusinaatioiksi. Kun tekoälyn tuottamaa sisältöä käytetään laajamittaisesti, hallusinaatioiden riski korostuu, varsinkin käyttäjien keskuudessa, jotka eivät ole tietoisia tämän kaltaisesta käyttäytymisestä. (Verma, Tran, Yusuf & Min 2023.)

Manipulatiiviset hyökkäykset. Kielimallit voivat olla alttiita manipuloiville syötteille, jotka saattavat johtaa virheisiin tai arkaluonteisen tiedon paljastumiseen. Esimerkiksi ChatGPT saadaan ohittamaan sille asetettuja varotoimenpiteitä syötteiden avulla (Liu yms. 2024).

Ennakkosenteet. Mallit voivat toistaa niiden koulutusaineistossa olevia ennakkoluuloja, mikä voi aiheuttaa epäoikeudenmukaisuutta ja syrjintää. Esimerkiksi, jos kielimalli on koulutettu historiallisella datalla, jossa ammatit kuten insinöörit ja lääkärit kuvataan useammin miehinä ja sairaanhoitajat sekä opettajat naisina, malli voi oppia ja toistaa näitä sukupuoleen perustuvia stereotyyppioita.

Johdonmukaisuuden puute. Saattaa olla haastavaa saada mallit seuraamaan annettuja ohjeita johdonmukaisesti ja ennakoitavasti. Mallit saattavat unohtaa ohjeistuksensa, jolloin niitä täytyy muistuttaa niiden tehtävästä ja noudatettavista säännöistä.

Haasteet, kuten alttius virheille ja loogisen päättelyn komplikaatiot mallien käytössä nousevat esiin erityisesti, kun kyse on mallien kyvystä tunnistaa milloin ja miten hyödyntää ulkopuolisia välineitä tai suorittaa monivaiheisia tehtäviä. Lisäksi suurten datamäärien käsittely ja monimutkaisten tietoaineistojen

analysointi tuovat mukanaan lisähaasteita. Kehotetekniikat pyrkivät vastaamaan juuri näihin haasteisiin, jotka ovat avainasemassa tekoälyn edistymisessä.

Haasteiden tunnistaminen ja niihin keinojen löytäminen on välttämätöntä mallien kehittämisen kannalta. Teknologian kehittyessä on ensiarvoisen tärkeää ymmärtää näiden mallien rajallisuudet ja parannusmahdollisuudet, mikä mahdollistaa entistä tehokkaampien ja luotettavampien tekoälyjärjestelmien kehittämisen.

7.3 Opinnäytetyön luotettavuus

Opinnäytetyössä käytettyjen tutkimusmenetelmien valinta on tehty huolellisesti, ja ne vastaavat tutkimuksen tavoitteita, mikä tukee työn luotettavuutta. Menetelmien soveltuvuus ja niiden läpinäkyvä esitystapa mahdollistavat tulosten toistettavuuden, joka on merkittävä osa tieteellisen työn luotettavuutta. Lisäksi työssä on viitattu huolellisesti kaikkiin käytettyihin lähteisiin, mikä mahdollistaa tietojen alkuperän tarkastamisen ja lisää työn uskottavuutta.

Opinnäytetyössä käsiteltyjen aihealueiden tiedot saattavat olla suppeita johtuen aiheen laaja-alaisuudesta ja tekoälyn tutkimusalan nopeasta kehityksestä. Ala on suhteellisen nuori ja teknologiset uudistukset tapahtuvat tiheään tahtiin, minkä seurauksena myös tutkimusmenetelmät kehittyvät ja laajentavat jatkuvasti ymmärrystämme kielimalleista ja niiden käyttäytymisestä.

7.4 Opinnäytetyön tuotosten tarkastelu ja jatkokehitysideat

Opinnäytetyössä keskityttiin kehoitteiden teorian tutkimiseen ja sen pohjalta kehitettiin oma GPT-malli, jonka luomisprosessi dokumentoitiin vaihe vaiheelta. Tämän GPT:n käyttötarkoitus oli ensisijaisesti demonstratiivinen, ja valinta perustui käytettävissä olevaan rajapintaan, jonka kautta GPT:n toiminnot oli mahdollista integroida.

Expenses-GPT osoittautui odotusten mukaiseksi niin generoitujen vastausten kuin toimintojenkin osalta. Se havainnollisti, kuinka GPT:n integroiminen ulkoisiin palveluihin voi tuoda merkittävää lisäarvoa ja avata uusia käyttömahdollisuuksia GPT:n kyvykkyyksille. Tämä demonstroi, kuinka tekoälyn ja erityisesti GPT-mallien avulla voidaan kehittää innovatiivisia ratkaisuja, jotka parantavat ja laajentavat digitaalisten palvelujen tarjontaa.

GPT-mallin jatkokehityksessä keskitytään sen toiminnallisuuden hienosäätöön perustuen käyttäjien antamaan palautteeseen. Mahdollinen kehityssuunta on GPT:n integrointi verkkosivustoihin käyttämällä OpenAI Assistant APIa. Tähän liittyen ehdotetaan oman React-komponentin kehittämistä, joka kykenee hankkimaan tietoa valitusta avoimesta rajapinnasta ja esittämään sen käyttäjille heidän toiveidensa mukaisesti. Tämän avulla voidaan esimerkiksi tarjota käyttäjille säädataa tai muita ajankohtaisia tietoja, hyödyntäen GPT:n kehoitteiden rakennusperiaatteita. Tämä lähestymistapa mahdollistaa GPT-mallin monipuolisemman käytön ja tehostaa sen soveltuvuutta erilaisiin digitaalisiin palveluihin.

LÄHTEET

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., Hesse, C., Sigler, E., Litwin M., Gray, Sc, Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D. 2020. Language Models are Few-Shot Learners. <https://doi.org/10.48550/arxiv.2005.14165>

Buruk, O. 2023. Academic Writing with GPT-3.5 (ChatGPT): Reflections on Practices, Efficacy and Transparency. <https://trepo.tuni.fi/handle/10024/154013>

Clarke, S., Milmo, D., Blight, G. 2023. How AI chatbots like ChatGPT or Bard work - visual explainer. The Guardian. Viitattu 12.3.2024. <https://www.theguardian.com/technology/ng-interactive/2023/nov/01/how-ai-chatbots-like-chatgpt-or-bard-work-visual-explainer>

How to Overcome LLM Training Challenges. n.d. Turing. Viitattu 1.4.2024 <https://www.turing.com/resources/llm-training-challenges#how-does-llm-training-work?>

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., Iwasawa, Y. 2023. Large Language Models are Zero-Shot Reasoners. <https://arxiv.org/abs/2205.11916>

Lane, H., Hapke, H.M., & Howard, C. 2019. Natural Language Processing in Action: Understanding, analyzing, and generating text with Python. Shelter Island, NY: Manning Publications.

Liu, Y., Deng, G., Xu, Z., Li, Y., Zheng, Y., Zhang, Y., Zhao, L., Zhang, T., Wang, K., & Liu, Y. 2024. Jailbreaking ChatGPT via Prompt Engineering: An Empirical Study. <https://doi.org/10.48550/arxiv.2305.13860>

Lähde, O. 2023. GPT-mallien hyödyntäminen ohjelmistokehityksessä. <https://www.theseus.fi/handle/10024/796749>

Niinivaara, N. 2023. GPT-algoritmin hyödyntäminen ohjelmistokehityksessä. <https://www.theseus.fi/handle/10024/812764>

OpenAI. 2024a. What are tokens and how to count them? Viitattu 14.2.2024. <https://help.openai.com/en/articles/4936856-what-are-tokens-and-how-to-count-them>

OpenAI. 2024b. Best Practices for Prompt Engineering. Viitattu 15.3.2024. <https://help.openai.com/en/articles/6654000-best-practices-for-prompt-engineering-with-the-openai-api>

Prompt Engineering Guide. n.d.-a. Prompt Elements. Viitattu 3.4.2024. <https://www.promptingguide.ai/introduction/elements>

Prompt Engineering Guide. n.d.-b. Tips. Viitattu 3.4.2024. <https://www.promptingguide.ai/introduction/tips>

Prompt Engineering Guide. n.d.-c. Zero-Shot Prompting. Viitattu 3.4.2024.
<https://www.promptingguide.ai/techniques/zeroshot>

Verma, S., Tran, K., Yusuf, A., & Min, G. 2023. Reducing LLM Hallucinations using Epistemic Neural Networks. <https://doi.org/10.48550/arxiv.2312.15576>

Wang, Z. J., Chakravarthy, A., Munechika, D., Chau, D. H. 2024. Wordflow: Social Prompt Engineering for Large Language Models. Viitattu 13.3.2024.
<https://arxiv.org/abs/2401.14447>

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., & Zhou, D. 2023. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. <https://doi.org/10.48550/arxiv.2201.11903>

What are Large Language Models? n.d. Viitattu 1.4.2024.
<https://www.ibm.com/topics/large-language-models>

Zamfirescu-Pereira, J.D., Wong, R. Y., Hartmann, B., Yang, Q. 2023. Why Johnny Can't Prompt: How Non-AI Experts Try (and fail) to Design LLM Prompts. <https://doi.org/10.1145/3544548.3581388>