

# Terapeut chatbot

Marcus Schanche, 18.11.2024

## BESKRIVELSE

**Forretningsmål:** Målet med dette prosjektet er å utvikle "Mindful Mike", en terapeut chatbot designet for å støtte individer i å håndtere sin mentale helse. Chatboten skal tilby tilgjengelig, umiddelbar mental helse-støtte mot en månedlig betaling. Målet er også å forbedre generell mental helse og senke listen for å snakke om seg selv og sitt liv.

### Eksisterende løsninger:

For øyeblikket gis mental helse-støtte gjennom terapiansamtaler ansikt til ansikt, hjelpetelefonlinjer og forskjellige nettbaserte terapi-plattformer. Uten maskinlæring utføres oppgaven typisk av utdannede terapeuter som møter pasienter for å gi støtte og råd.

**Bruk av løsningen:** "Mindful Mike" vil integreres i en mental helse-app eller plattform og vil være tilgjengelig via nett- og mobilgrensesnitt. Brukere vil samhandle med chatboten gjennom tekstbaserte samtaler.

## METRIKKER

For at prosjektet skal betraktes som en suksess, må visse minimumskrav innen både forretningsytelse og teknisk ytelse oppfylles. Her er en oversikt over de metrikker som vil bli brukt og deres sammenheng med forretningsmålene.

### Forretningsmetrikker:

#### Engasjementmetrikker:

- **Antall økter per bruker:** For å sikre at brukerne finner verdi i chatboten og fortsetter å bruke den, bør hver bruker i gjennomsnitt gjennomføre minst tre økter per uke. Dette vil indikere at chatten er nyttig og engasjerende.
- **Gjennomsnittlig økttid:** En gjennomsnittlig økttid på minst fem minutter vil indikere at brukerne er engasjert og får tilstrekkelig hjelp fra chatboten.

#### Brukertilfredshet:

- **Brukervurderinger og tilbakemeldinger:** Chatboten bør motta en gjennomsnittlig vurdering på minst 4 av 5 stjerner. I tillegg bør minst 80% av tilbakemeldingene være positive eller indikere tilfredshet med chat-opplevelsen.

#### Retensjonsgrad:

- **Prosentandel av tilbakevendende brukere:** For å sikre langsiktig bærekraft og brukertrofasthet, bør minst 50% av brukerne vende tilbake og bruke chatboten på nytt innen en måned etter første bruk.

Disse forretningsmetrikker sørger for at chatboten ikke bare er teknisk kapabel, men også verdifull og tilfredsstillende for brukerne, noe som er essensielt for å oppnå forretningsmålene.

### **Maskinlærings- og software-metrikker:**

#### **Treffsikkerhet (Accuracy):**

- **Treffsikkerhet (Accuracy) på brukerinteraksjoner:** Modellens treffsikkerhet bør være minst 85%. Dette betyr at 85% av chatbotens svar skal være korrekte og relevante, noe som sikrer at brukerne får pålitelig og nyttig informasjon.

#### **Latency:**

- **Responstid:** Chatboten skal ha en gjennomsnittlig responstid på under 2 sekunder. Dette sikrer en jevn og rask samtaleflyt, noe som er avgjørende for brukertilfredshet.

#### **Throughput:**

- **Antall betjente brukere per tidsenhet:** Systemet skal kunne håndtere minst 100 samtidige brukere uten merkbare ytelsesforringelser. Dette er nødvendig for å sikre at chatboten kan betjene et stort antall brukere samtidig, spesielt under toppekstimer i bruk.

#### **Mean Squared Error (MSE):**

- **Kvadratisk avvik i sentimentanalyse:** Hvis sentimentanalyse brukes for å forstå brukerens emosjonelle tilstand, bør MSE være lav for å sikre nøyaktige prediksjoner av brukerens følelser. Lav MSE betyr at chatboten bedre kan tilpasse svarene basert på brukernes sinnsstemning.

Denne kombinasjonen av metrikker sikrer at løsningen ikke bare fungerer fra et teknisk perspektiv, men også leverer en høy kvalitet på brukeropplevelsen. Ved å kontinuerlig overvåke og optimalisere disse metrikker sikrer vi at chatboten "Mindful Mike" oppfyller både de tekniske kravene og forretningsmålene.

## **DATA**

**Datainnsamling og behandling:** Data samles inn gjennom interaksjoner mellom brukerne og chatboten i appen eller plattformen. Selv om GPT-4 er pre-trent, vil det være nødvendig å samle inn en betydelig mengde data for å kunne trene og finjustere modellen. Dette gjøres for å tilpasse chatboten til spesifikke brukstilfeller og for å forbedre nøyaktigheten og relevansen av svarene. Gjennom kontinuerlig datainnsamling og analyse oppnås en optimal ytelse og en brukervennlig opplevelse.

**Dat typer og merking:** Den primære datakilden vil være tekstbaserte samtaler mellom brukerne og chatboten. For å forbedre modellens nøyaktighet og ytelse, annoteres disse dataene for å indikere om chatbotens svar var riktige eller hjelpsomme, basert på brukerfeedback. Dette gir innsikt i hvordan modellen presterer og hvilke områder som kan forbedres.

**Personvern hensyn:** Beskyttelse av brukernes personvern er av største viktighet, og derfor må strenge retningslinjer for datainnsamling og lagring følges. Der det er mulig bør dataene anonymiseres for å beskytte individet. Dette ikke bare fordeler forhold til å opprettholde brukernes tillit, men også overholdelse av relevante personvernlover og forskrifter.

**Datarepresentasjon:** For å gjøre tekstdataene egnet for maskinlæring, må de renses og forhåndsbehandles nøye. Dette innebærer prosesser som feature engineering og skalering, som er nødvendige for å forbedre modellens ytelse. Rensede og skalerte data gir modellen et bedre grunnlag for å lære og forstå variablene som påvirker ytelsen.

Ved å fokusere på alle disse aspektene av data, sikres en kontinuerlig forbedring av chatbotens evner. Datainnsamlingen gjør det mulig å finjustere og tilpasse modellen for å møte spesifikke behov, mens grundige personverntiltak beskytter brukerne. Samlet sett gir dette en sterkere, mer pålitelig og responsiv chatbot som kan møte brukernes krav og forventninger på en effektiv og sikker måte.

## MODELLERING

Vi starter med enkle maskinlæringsmodeller som logistisk regresjon og beslutningstrær for å etablere en grunnleggende ytelse. Dette gir oss en baseline-ytelse som vi kan sammenligne mot mer avanserte modeller senere. For å sette en baseline ser vi også på enkle regelbaserte systemer og resultater fra lignende prosjekter, samt sammenligner med menneskelig ytelse der det er relevant.

For å forbedre modellene vil vi nøye undersøke tilfellene der modellene tar feil. Dette gir innsikt i hvilke aspekter av dataene som kan være problematiske. Vi analyserer også "feature importance" for å se hvilke dataelementer som mest påvirker modellens avgjørelser. Denne informasjonen brukes til å justere dataene, forbedre modellens innstillinger og eventuelt legge til flere relevante funksjoner, slik at modellen gir mer nøyaktige og pålitelige resultater.

## DEPLOYMENT

Modellene vil bli satt i drift ved å integrere dem i applikasjonen eller systemet hvor brukerne kan interagere med dem. Dette innebærer å sette opp en server eller en skytjeneste som kjører modellen og gjør den tilgjengelig via et API. Prediksjonene fra modellen vil bli brukt til å besvare spørsmål, gi anbefalinger eller utføre andre oppgaver som modellen er trent til. For monitorering og vedlikehold av systemet vil vi sette opp logger og overvåkingsverktøy for å spore modellens ytelse i sanntid. Dette inkluderer å måle nøyaktighet, svartid og brukertilfredshet. Ved eventuelle avvik i ytelsen kan vi raskt identifisere og rette opp problemer. Vi vil også implementere automatiske alarmer som varsler utviklingsteamet ved uvanlige aktiviteter eller feil.

Planer for forbedring av systemet etter at det er satt i drift inkluderer kontinuerlig innsamling av nye data fra brukernes interaksjoner, som deretter kan brukes til å rettene og forbedre

modellen. Vi vil også løpende evaluere nye maskinlæringsmetoder og teknologier som kan integreres for å optimalisere ytelse. Brukerfeedback vil også være en viktig kilde for å forstå hvor systemet kan bli bedre, og vi vil regelmessig oppdatere modellen basert på disse innsiktene.

## REFERANSER

### Artikler:

Model Improvement and User Feedback:

<https://dl.acm.org/doi/10.1145/1060745.1060797>

Mean squared error:

[https://en.wikipedia.org/wiki/Mean\\_squared\\_error](https://en.wikipedia.org/wiki/Mean_squared_error)

### Nettsteder:

<https://platform.openai.com/docs/>

<https://github.com/openai/openai-python>

<https://www.gradio.app/guides/quickstart>

<https://github.com/openai/openai-python>