

howest
hogeschool

Thuisproject

Ibe Demortier

Table of Contents

- Verslag.....3**
 - Groepsleden + verdeling.....3
 - Bespreking dataset.....3
 - Bespreking zoekopdrachten.....4
 - Schema5
 - Overzicht.....6
 - Moeilijkheden7

Verslag

Groepsleden + verdeling

Ibe:

- Preprocessen data (~3-4uur)
- Programmeren front-end (~6uur)
- Programmeren back-end (~8uur)
- Documentatie (~3uur)

Bespreking dataset

<https://www.kaggle.com/datasets/unsdsn/world-happiness/data>

De dataset omvat de gelukscores van verschillende landen gedurende de jaren 2015 – 2019. De belangrijkste variabelen omvatten de gelukscore zelf, socio-economische indicatoren zoals de BBP, en nog veel meer.

Na een analyse van de data valt er aan te tonen dat er grote verschillen zijn tussen de landen en over de tijd. Bij sommige landen is er een grote schommeling in de jaren te zien terwijl dat dan bij andere landen zeer consistent blijft. In het algemeen valt er wel te besluiten dat de landen met een hoger BBP ook een hogere gelukscore hebben. Landen met een lager BBP tonen dan ook een lagere gelukscore.

Deze bevinden zijn allemaal zeer mooi te zien op grafieken. Zo valt er ook mooi te zien dat de gelukscore ook zeer hard kan verschillen van regio tot regio. Zo ligt de gemiddelde gelukscore in West-Europa, Noord-Amerika en Australië veel hoger dan in de rest van de wereld.

Er waren ook wel enige gebreken aan de dataset zoals dat de data over de jaren niet consistent is. Sommige kolommen zijn niet over elk jaar beschikbaar of soms wordt er een nieuwe toegevoegd. Ook de namen van de kolommen over de verschillende jaren zijn veranderd wat er voor zorgde dat er eerst redelijk wat preprocessing moest gedaan worden.

In conclusie biedt de dataset wel een waardevolle bron voor onderzoek naar het welzijn op wereldschaal. De dataset kan zeer leerrijk zijn en er valt nuttige informatie uit te halen. Het geeft ook een duidelijke kijk op de wereld om te zien waar het beter gaat dan op andere plaatsen.

Bespreking zoekopdrachten

Ik heb in het project 4 zoekopdrachten uitgewerkt.

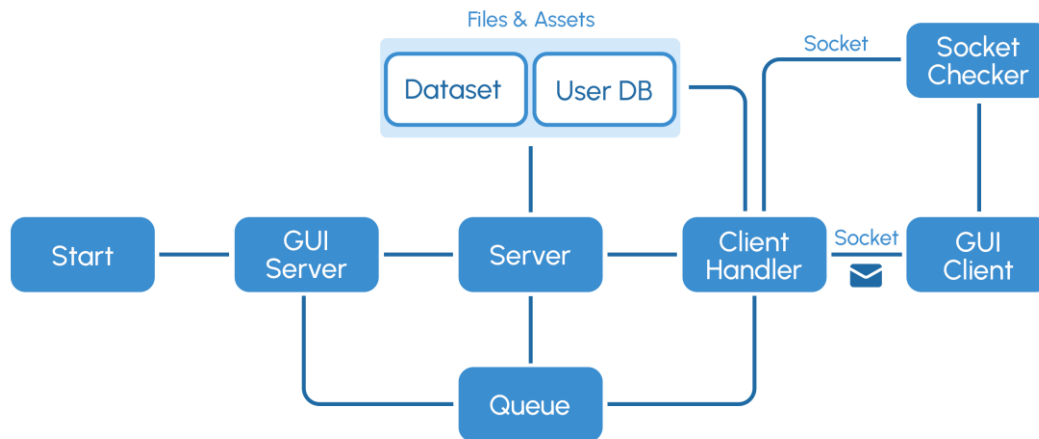
De eerste zoekopdracht geeft alle landen tussen een bepaalde score range die je kan meegeven in de client van de gui. Hierdoor kan je duidelijk zien welke landen waar ongeveer liggen in de score range en hoe goed sommige landen het hebben of hoe slecht. Voor het invullen van de score range heb ik gekozen voor 2 numeric up-down boxen. Deze heb ik gekozen omdat deze het best gemaakt zijn om cijfers in te geven en zorgt voor zo voor zo min mogelijke fouten. Ook is er voor gezorgd dat je zelf mag kiezen of het kleinste of grootste cijfer eerst staat.

De tweede zoekopdracht geef De score weer over de 5 jaar van een bepaald land dat je zelf kan doorgeven. Ook wordt er hier een plot van getoond om mooi het verloop van de score te zien over de 5 jaar. Met deze plot is het zeer duidelijk of een land gelukkiger wordt, minder geluk, of hetzelfde blijft. Voor het doorgeven van de naam van het land maak ik gebruik van een combobox. Dit heb ik gekozen omdat er hierdoor geen schrijffouten kunnen gemaakt worden zoals dit zou kunnen bij een entrybox. Zo kan je ook alle mogelijke landen onder elkaar zien en zelf zoeken achter het land waar je de info van wil.

De derde zoekopdracht is een BBP range ingeven om zo alle landen in deze range te zien. Hierdoor kan je duidelijk zien welke landen tussen welke BBP score liggen. Hieruit kan je afleiden hoe sommige landen het doen op vlak van BBP. Om de range van de BBP in te geven heb ik ook weer gekozen voor een numeric up-down box. Hierdoor zorg ik er ook voor dat je niet buiten de range van de BBP gaat zoeken.

De vierde en laatste zoekopdracht maakt het mogelijk om de gelukscore van 2 landen met elkaar te vergelijken. De gelukscore van de 2 landen over de 5 jaar komen mooi in 1 venster te staan en zijn zo makkelijk te vergelijken met elkaar. Voor het invoeren van de landen heb ik een combobox gebruikt. Dit voorkomt typfouten en maakt het makkelijk om een land uit de lijst te kiezen.

Schema



Overzicht van de Server-Client Architectuur

- `start.py`: Dit script start een instantie van `gui_server.py`, dat een GUI implementeert met behulp van Tkinter. Het doorgeven van een Tkinter-object aan `gui_server.py` resulteert in het tonen van een GUI-venster.
- `gui_server.py`: Deze module, die overerft van Tkinter, toont een GUI-venster en fungeert als een interface voor de gebruiker. Het kan rechtstreeks methoden van `server.py` en `clienthandler.py` aanspreken, maar voert geen logica uit.
- `server.py`: Hier wordt de socket gecreëerd en wordt geluisterd naar inkomende verbindingen. Voor elke verbonden client maakt het een `clienthandler`-instantie aan, die de socket en de wachtrij (queue) krijgt om te communiceren met de client.
- `clienthandler.py`: Dit bestand bevat de logica voor het verwerken van inkomende berichten van de client en het doorsturen ervan naar de server. De `clienthandler` communiceert met de GUI van de client via JSON-formaat, waarbij custom objecten met properties worden omgezet naar JSON en vice versa.
- `gui_client.py`: Deze module initialiseert een GUI-client die verbinding maakt met dezelfde socket waar de server naar luistert. Het ontvangt berichten van de server via de socket en toont ze aan de gebruiker.

Communicatie tussen Componenten:

- De server en de `clienthandler`-instanties communiceren met de GUI via een wachtrij (queue). Berichten van de client worden rechtstreeks op de wachtrij geplaatst, en de server luistert continu naar deze wachtrij om de berichten door te geven aan `gui_server.py` voor visuele weergave.

Overzicht

- Data preprocessing

Ik heb de data preprocessing succesvol kunnen uitvoeren op de dataset en perfect de dataset kunnen verkrijgen zoals ik dit wou. Ook heb ik 5 plots kunnen maken die mooi wat info tonen over de dataset en waar er ook het een en het ander uit te leren valt. Ze tonen mooi aan waar de data handig voor is.

- Client-server connectie

Ik heb een client-server connectie kunnen opzetten waarmee er kan gecommuniceerd worden tussen de server en de client. Dit allemaal gebeurd ook via een queue.

- Client

Bij het opstarten van de client zal er eerst een venster tevoorschijn komen waarin wordt gevraagd om in te loggen of een account te creëren. Eens dat dit voltooid is zal de gui van de client tevoorschijn komen. Hiermee kan je de verschillende zoekopdrachten uitvoeren. Ik heb alle 4 de gevraagde zoekopdrachten kunnen maken en ik kan ook een plot tonen. Bij het opstarten van de client zal er ook gewacht worden tot wanneer de server effectief aan het draaien is dit voorkomt error's.

- Server, clienthandler en GUI-server

De server kan berichten en commando's van de server ontvangen en hiermee de zoekopdrachten gaan uitvoeren om dan de data terug te sturen naar de client. Op de GUI van de server worden alle belangrijke acties in een logvenster getoond, is er een venster te zien waarin alle verbonden clients te zien zijn. Kan je per user in de dataset kijken hoeveel keer elke user elke zoekopdracht heeft gebruikt. En is er ook per zoekopdracht te zien hoeveel keer deze is gebruikt.

Hieruit kan er besloten worden dat mijn project aan alle gevraagde criteria voldoet en dat het ook werkt naar mijn behoren.

Moeilijkheden

In het begin had ik het moeilijk met de logica van het project in mijn hoofd krijgen. Dan heb ik eerst eens rustig de logica opgeschreven hoe het hoort te werken en dan ben ik zo tot deze werkende logica om altijd naar de binnenkomende commando's te kijken en zo de juiste zoekopdracht te laten uitvoeren.

Eens de logica volledig af was en ik de zoekopdrachten kon opvragen ben ik begonnen met de info tonen op de GUI van de server. Hier had ik eerst wat moeite met een inlogscherf maken voor de client. Dit is mij uiteindelijk dan toch gelukt door rustig te beredeneren hoe het zou moeten werken en heb zo dan een users csv gemaakt wat het makkelijk maakt om de gebruikers bij te houden.

howest
hogeschool