An Phríomh-Oifig Staidrimh
Central Statistics Office

The Synergy of R and Generative AI in Statistics

uRos2024
Athens

Introduction

- Vytas Vaiciulis
- Central Statistics Office Ireland
- Vytas Vaiciulis, Central Statistics Office Ireland, Lead of the Analytics & AI Innovations Section

R for multi purpose use

- Serves as a powerful tool for data manipulation, analysis, and visualization.
- Played a central role in building the Code Translator application.
- Demonstrates versatility by translating other programming languages into R, highlighting its broad utility.
- Combines development and execution, showcasing adaptability in real-world projects.
- Recognized as a premier language for statistics, empowering advanced analytics and data science.

Code translator explanation

This application facilitates translation between programming languages, such as using it to convert code from another language to R.

The application uses the OpenAI API for code translation. However, alternative LLMs from other providers could be integrated with some adjustments to the source code, which is beyond the scope of this presentation.

Code translator r libs

```
library(shiny)
library(openai)
library(shinythemes)
library(waiter)
library(shinyAce)
library(future)
library(promises)
library(regex)
```

Prompt Engineering examples

Lazy prompt

Do R code

Basic prompt

Translate SAS code into R code

Good prompt

Translate SAS code into R equivalent using Tidyverse, comment the code, and load the library.

Better prompt

Translate SAS proc sql into R equivalent using dplyr, comment the code, and load the library.

Added **QuillBot** (A.I driven) as chrome extension to help with semantics.

Guardrail prompt for the translator

"You are a SAS to R transcompiler. I will give you SAS code and you turn it into R code that achieves the equivalent result. Do not ask the R code to print anything unless it is necessary in the SAS code. Please start the response with the equivalent R code would be in R. Briefly comment the code, and in the code comments briefly mention any important assumptions, only if any were made. Do not respond further after giving the R code. Always use Tidyverse packages where possible."

Code translation example

SAS code

Prompt

```
proc export data=mydata
outfile='mydata.csv'
dbms=csv replace;

run;
```

R code

Completion

```
write.csv(mydata, file = "mydata.csv" ,
row.names = FALSE, sep = ",")
```

Explainer Prompt Testing

To examine the efficiency, accuracy and consistency of the prompt given to OpenAI's API, we have conducted a series of "Trial Runs". These runs are split up into 3 categories:

Small: 2 – 10 lines of simple R code.

Medium: 10 – 25 lines of R code.

Large: 25 + lines of R code.

Translator open-source (public)

- Code-Transcompiler-using-LLMs-public
- MIT license
- Docker
- https://github.com/CSOIreland/Code-Transcompiler-using-LLMs-public

Conclusion

- **Code translation:** Facilitating translation between any two programming languages.
- **Standardisation:** Paving the way for unprecedented standardisation across various platforms and languages.
- **Code simplification:** Reducing and simplifying code complexity for easier comprehension and maintenance.
- **Enhanced explanations:** Providing clearer and more detailed explanations of translated code sections.
- **Efficiency gains:** Offering significant efficiency improvements, especially when handling and analysing big data.
- **Language to code creation:** Translating verbal instructions directly into code.

Credit to contributors

Interns/Graduates

- Sean Kelly
- Minh Quach
- Eva Leahy
- Sean Browne

Permanent staff

- Peter Marsh
- Vytas Vaiciulis
- Vinicus Andrade
- Callum Wilson