Exploring the Use of LLMs to Generate Contextualized Worked Examples From Programming Patterns

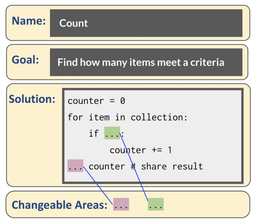
Yoshee Jain, Katie Cunningham

CS STARS
STUDENT AMBASSADORS/RESEARCH SCHOLARS

Introduction

What is a pattern?

A programming pattern is a piece of code common to programs from a particular application area (e.g., text parsing) that achieves a specific goal in that context. It has a *name and goal*, a *solution*, and *highlighted changeable areas*.



Why is pattern-oriented instruction important?

Pattern-oriented instruction is promising for learners because it supports them to focus on code's purpose and applications instead of syntax.

What are worked examples and why are they effective?

An example with the problem statement and a step-by-step procedure for solving the problem. Such example reduce students' cognitive load when learning new and unfamiliar computing topics.

Why is learning using contextualized examples effective?

Learning about programming with examples that are contextualized with stories relevant to students' interests is motivating.

If students have access to a large number of contextualized worked examples, then they will be able to learn computing topics more effectively!

Why is this challenging to incorporate in practice?

Creating contextualized examples that contain multiple patterns relevant to recent lessons is a human-dependent task that is tedious and time-consuming.

We explore if LLMs are capable of compiling patterns to create contextualized worked examples when given maximum support:

(1) three patterns with a natural language description of their purpose (goals),

(2) a code example,

(3) and the correct order of arranging the patterns in an example.

Why is incorporating all types of support in the prompt not sufficient?

(1) It would still be tedious for instructors to incorporate all types of support in their query.

(2) Patterns have been identified in a variety of ways in the literature; these patterns may not always contain both the goals and the code.

(3) Organizing a list of patterns logically would be cognitively demanding.

[1] K. Cunningham et al. "Avoiding the Turing Tarpit: Learning Conversational Programming by Starting from Code's Purpose."

RQ: To what extent can LLMs compile patterns to create contextualized worked examples?

Name of Pattern	Goal of Pattern
Create DataFrame from Dictionary	Create a sample DataFrame from some initial data defined in a dictionary
Read CSV into a DataFrame	Read a CSV file into memory as a DataFrame
Filter rows in a column based on a condition	Filter the rows where the value is less than a specified threshold for a specific column in the DataFrame
Group by and aggregate	Take a DataFrame, group by an attribute, and aggregate on one column to find a sum
Sample	Sample ***n*** rows randomly from the data frame
Merge Dataframes	Merge or concatenate multiple DataFrames together
Calculate the mean of a column	Calculate the mean across all values in a column
Plot a histogram	Plot the histogram of a specified column
Export a DataFrame to a CSV file	Create output.csv on disk in the current directory, which stores the contents of DataFrame

Methodology

(1) Prompted ChatGPT with patterns from Pandas to generate contextualized examples for instruction.

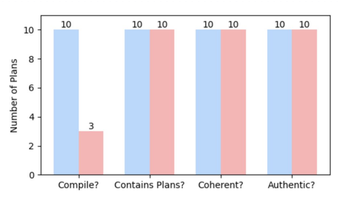
(2) Created a baseline dataset quantifying ChatGPT's performance when given maximum support

(3) Evaluated the quality of the generated examples when given varying amounts of support across four metrics in comparison to this baseline.

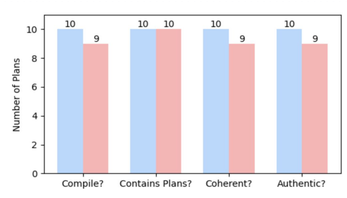
Quality Metrics for Evaluation

(1) Syntactic Validity
(2) If They Contain The Requested Patterns
(3) If The Contextualization Is Coherent
(4) Alignment Of Code With Standard Practice

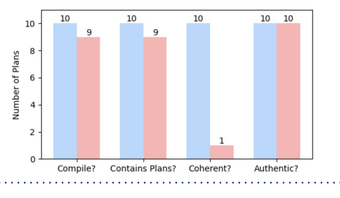
ChatGPT can generate quality examples when the pattern code is not explicitly included in the prompt but is provided as a "background guide" in a RAG-based implementation.



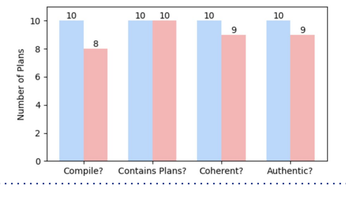
LLMs can generate quality contextualized worked examples even when patterns are not described in natural language in the prompt.



When prompted with many patterns, the examples generated by ChatGPT need more instructor editing to ensure quality.



LLMs can arrange patterns logically when creating a contextualized example, even if they are not given in a meaningful order.



We suggest that a RAG-based one-shot prompting approach including only the patterns' goals will give instructors the most flexibility to create contextualized examples using many patterns in any order!

Future Work

(1) Explore the impact of interactions between the given amount of support and the quality of the generated examples.

(2) Explore the generation of worked examples in more advanced application areas, such as web development with Django or machine learning with Pytorch.

Acknowledgements

I would like to thank Prof. Katie and the TRAILS Lab members for their support throughout the project. I also appreciate funding and support from the CS STARS program.