

Generative AI and Variability A Research Vision

Sandra Greiner Klaus Schmid² Thorsten Berger³ Sebastian Krieter⁴ Kristof Meixner⁵

Software Engineering Group
University of Bern, Switzerland

Februray, 9th 2024

VaMoS '24 in Bern

²University of Hildesheim ³Ruhr University Bochum ⁴University of Ulm ⁵TU Vienna

Current State of Managing Variable-Intensive Software

SOTA: Large (Evolving) Variability-Intensive Software

The screenshot shows the GitHub repository page for `nasa/fprime`. The repository is public and has 1.2k forks and 9.3k stars. It is categorized as a flight software and embedded systems framework. The repository includes a list of submodules and recent updates.

Module	Description	Update Time
<code>.github</code>	lestarch: removing permissions-intensive rep...	6 months ago
<code>Autocoders</code>	Update check-spelling to v0.0.20 (#1583)	7 months ago
<code>CFDP</code>	mstarch: all cmake is now targets, deploymen...	2 years ago
<code>Drv</code>	Add sys/time.h include to IpSocket for musl c...	9 months ago
<code>Fpp</code>	Revise Ref topology model	2 years ago
<code>Fw</code>	Update check-spelling to v0.0.20 (#1583)	7 months ago
<code>Os</code>	Only support set_cpu_affinity with glibc (#1517)	9 months ago
<code>RPI</code>	Latest ActiveRateGroup updates (#1510)	9 months ago
<code>Ref</code>	Update/ref int tests (#1479)	9 months ago
<code>STest</code>	Revise Deframer design and implementation (...)	last year
<code>Svc</code>	Update check-spelling to v0.0.20 (#1583)	7 months ago
<code>Utils</code>	Use identity comparison with singleton (#1501)	9 months ago
<code>ci</code>	lestarch: initial implementation of RPI integrat...	2 years ago
<code>cmake</code>	Update check-spelling to v0.0.20 (#1583)	7 months ago
<code>config</code>	Latest ActiveRateGroup updates (#1510)	9 months ago
<code>docs</code>	Update check-spelling to v0.0.20 (#1583)	7 months ago
<code>.clang-tidy</code>	Replace NULL with nullptr (#1049)	2 years ago
<code>.dockerignore</code>	Fix #1215 Remove incorrect references to non...	2 years ago
<code>.gitignore</code>	Fix #1215 Remove incorrect references to non...	2 years ago

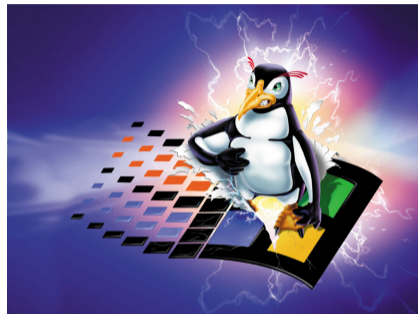
About
F⁺ - A flight software and embedded systems framework

Tags: raspberry-pi, components, real-time, framework, embedded, cpp, nasa, embedded-systems, flight, spaceflight, object-oriented-programming, flight-software, fprime

Stats: 9.3k stars, 1.2k forks, 258 watching

Releases: 15 releases, latest: `Release v3.1.1: fprime...` (Latest) on Aug 19, 2022

Contributors: 116 contributors



feature identification
& location

constraint extraction

variability modeling

feature identification
& location

annotative approaches

constraint extraction

feature-oriented,
transformative

variability modeling

feature identification
& location

annotative approaches

variant
sync & integration

constraint extraction

feature-oriented,
transformative

...

variability modeling

feature identification
& location

annotative approaches

variant
sync & integration

constraint extraction

feature-oriented,
transformative

...

language-specific, hard to analyze large systems, and to maintain consistently ...

variability modeling

feature identification
& location

annotative approaches

variant
sync & integration

constraint extraction

feature-oriented,
transformative

...

language-specific, hard to analyze large systems, and to maintain consistently ...

still insufficient genericity and automation

Business / Tech

A year after ChatGPT's release, the AI revolution is just beginning



By Catherine Thorbecke, CNN

8 minute read · Updated 10:32 AM EST, Thu November 30, 2023



Business / Tech

A year after ChatGPT's release, the AI revolution is just beginning



By Catherine Thorbecke, CNN

© 8 minute read · Updated 10:32 AM EST, Thu November 30, 2023

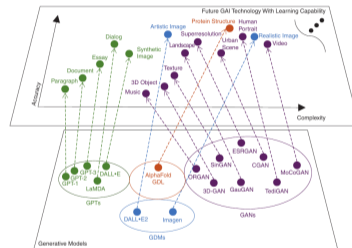


FIGURE 1. The GAI landscape: generative models and artifacts.

Business / Tech

A year after ChatGPT's release, the AI revolution is just beginning

By Catherine Thorbecke, CNN
 8 minute read · Updated 10:32 AM EST, Thu November 30, 2023



786

J.J. Cámara et al.

Prompt	UML Class diagram	PlantUML
I would like to write a class diagram in PlantUML in which a Videoclub rents movies. The Videoclub has customers, and both the Videoclub, the movies and the customers must have names.		<pre>classDiagram class Videoclub { - name: String } class Movie { - name: String } class Customer { - name: String } Videoclub "*" -- "*" Customer : has Videoclub "*" -- "*" Movie : rents @endclass</pre>

Fig. 2 Prompt used to ask ChatGPT to generate a UML class diagram of a video club system, and the resulting model

```
classDiagram
    class Videoclub {
        - name: string
        + getName(): string
        + setName(name: string): void
    }
    class Movie {
        - name: string
        + getName(): string
        + setName(name: string): void
    }
    class Customer {
        - name: string
        + getName(): string
        + setName(name: string): void
    }
    class Rental {
        - customer: Customer
        - movie: Movie
        - rentedDate: Date
        - returnedDate: Date
        + getCustomer(): Customer
        + setCustomer(customer: Customer): void
        + getMovie(): Movie
        + setMovie(movie: Movie): void
        + getRentedDate(): Date
        + setRentedDate(rentedDate: Date): void
        + getReturnedDate(): Date
        + setReturnedDate(returnedDate: Date): void
    }
    Videoclub --> Rental
    Movie --> Rental
    Customer --> Rental
```

Fig. 3 Another model generated by ChatGPT in response to exactly the same prompt, but in a different session

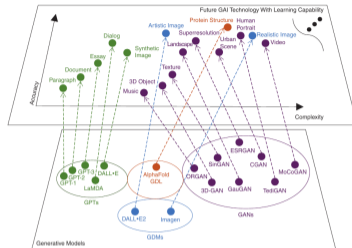


FIGURE 1. The GAI landscape: generative models and artifacts.

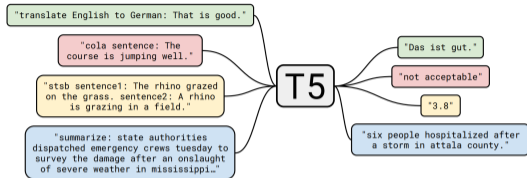
On Programming Variability with Large Language Model-based Assistant

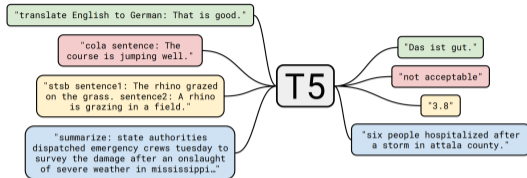
The screenshot shows a web interface for customizing a cat image. It features a 'Customization' section with four main categories: 'Ears, tail, and whisker', 'Body', 'Head and paws', and 'Ear and eyes (scale)'. Each category has several dropdown menus and sliders. Below the customization options, there are two images of a cat: 'vs original' and 'cat'.

Figure 1: End-user customization of Cat TikZ code

**What does it need from research
to support maintaining
variability-intensive software systems
with Generative AI?**

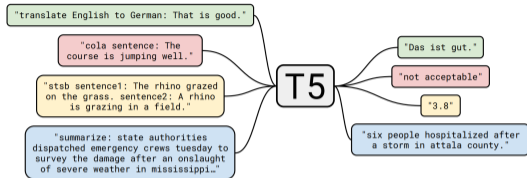
Background





The basic idea is to use a large language corpus to train a neural network to learn the language, by hiding part of the text and asking the network to guess the missing parts.

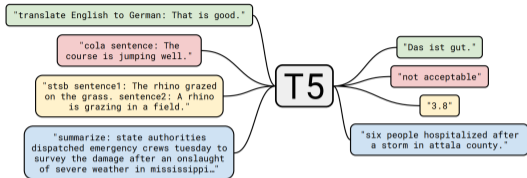
Generative AI – Large Language Models



The basic idea is to use a large language corpus to train a neural network to learn the language, by hiding part of the text and asking the network to guess the missing parts.

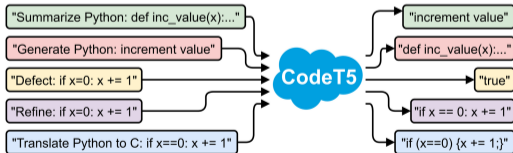
large pre-trained transformer models:
convert *input* text into *output* text
(encoder-decoder)

Generative AI – Large Language Models



The basic idea is to use a large language corpus to train a neural network to learn the language, by hiding part of the text and asking the network to guess the missing parts.

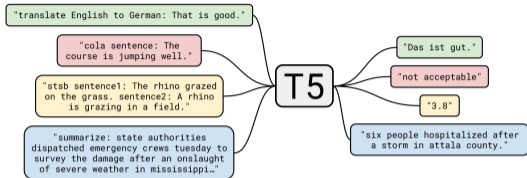
large pre-trained transformer models:
convert *input* text into *output* text
(encoder-decoder)



Generative AI for SE: pre-trained on several programming languages
defect & clone detection
code summarization
code translation,

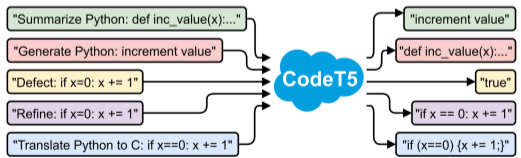
...

Generative AI – Large Language Models



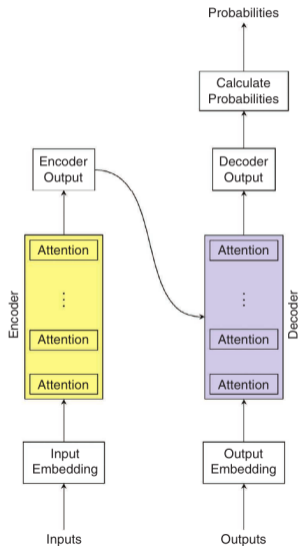
The basic idea is to use a large language corpus to train a neural network to learn the language, by hiding part of the text and asking the network to guess the missing parts.

large pre-trained transformer models:
convert *input* text into *output* text
(encoder-decoder)

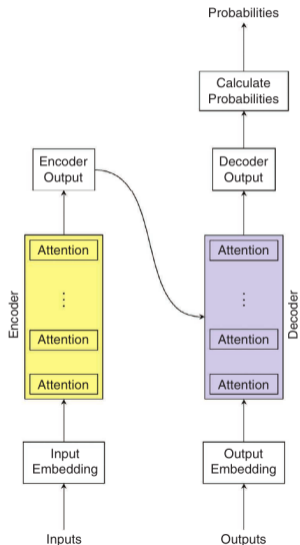


Generative AI for SE: pre-trained on several programming languages
defect & clone detection
code summarization
code translation,

...

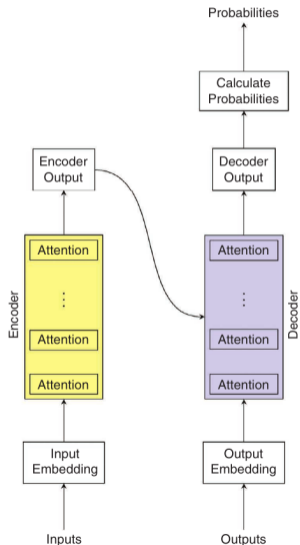


context window defines size of input and output (in tokens)



context window defines size of input and output (in tokens)

typical sizes: 8k-32k



context window defines size of input and output (in tokens)

typical sizes: 8k-32k

Further tuning through hyperparameters (temperature, learning rate, ...)

**What does it need from research
to support maintaining
variability-intensive software systems
with Generative AI?**

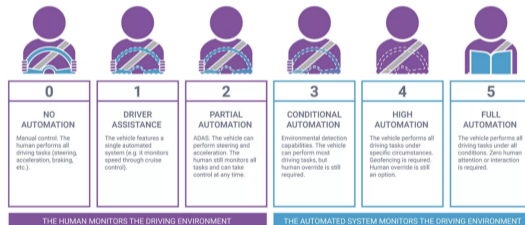
Idea

How can we assess success of Gen-AI-enabled development of variability-intensive software?

How can we assess success of Gen-AI-enabled development of variability-intensive software?

SYNOPSIS[®]

LEVELS OF DRIVING AUTOMATION



How can we **classify** and assess progress
in using Generative AI for variability-aware SE ?

How can we **classify** and assess progress
in using Generative AI for variability-aware SE ?

Scale information amount processed in one step

How can we **classify** and assess progress
in using Generative AI for variability-aware SE ?

Scale information amount processed in one step

Task Type analytical vs. adaptive, to a productive end*

How can we **classify** and assess progress
in using Generative AI for variability-aware SE ?

Scale	information amount processed in one step
Task Type	analytical vs. adaptive, to a productive end*
Quality	level of reliability: random, junior developer, senior developer, even better

How can we **classify** and assess progress
in using Generative AI for variability-aware SE ?

Scale	information amount processed in one step
Task Type	analytical vs. adaptive, to a productive end*
Quality	level of reliability: random, junior developer, senior developer, even better
Automation	level of manual intervention

*not a specific variability-intensive task

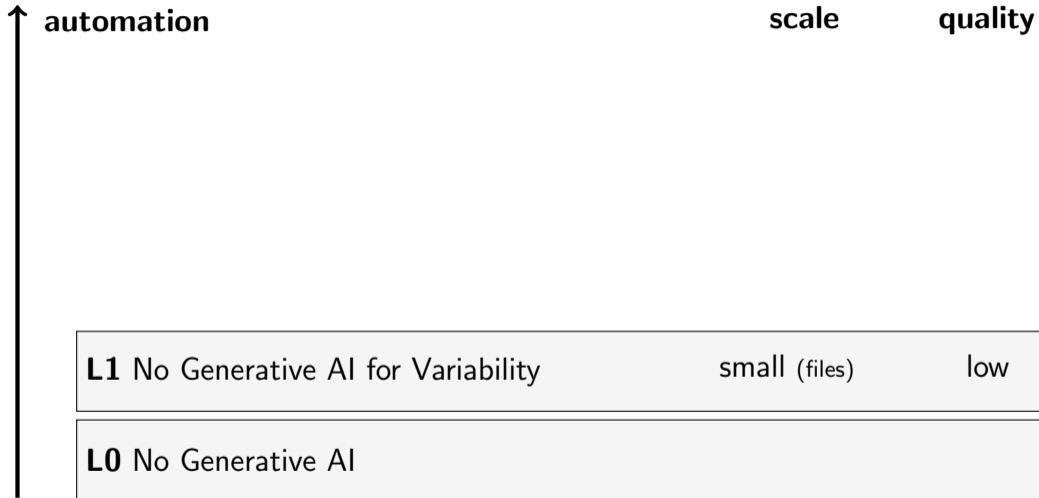


automation

scale

quality

L0 No Generative AI



The diagram shows a 2x3 grid of automation levels. A vertical arrow on the left points upwards, indicating increasing automation. The columns are labeled 'automation', 'scale', and 'quality'. The rows represent different levels of generative AI usage.

	automation	scale	quality
	L1 No Generative AI for Variability	small (files)	low
	L0 No Generative AI		

automation	scale	quality
L2 Basic Generative AI for Variability	small (project)	low
L1 No Generative AI for Variability	small (files)	low
L0 No Generative AI		

automation	scale	quality
L3 Advanced Generative AI for Variability	multiple (projects)	high
L2 Basic Generative AI for Variability	small (project)	low
L1 No Generative AI for Variability	small (files)	low
L0 No Generative AI		

↑ automation	scale	quality
L4 Full Generative AI for Variability	unlimited	expert
L3 Advanced Generative AI for Variability	multiple (projects)	high
L2 Basic Generative AI for Variability	small (project)	low
L1 No Generative AI for Variability	small (files)	low
L0 No Generative AI		

How can we achieve increase?

How can we achieve increase?

Increase Scale

How can we achieve increase?

Increase Scale

generic solutions

SE solutions

SPLE solutions (partitioning of large software projects)

How can we achieve increase?

Increase Scale

generic solutions

SE solutions

SPLE solutions (partitioning of large software projects)

Increase Task

towards reaching productive usage

How can we achieve increase?

Increase Scale

generic solutions

SE solutions

SPLE solutions (partitioning of large software projects)

Increase Task

towards reaching productive usage

when and how integrate existing tools

ensure traceability

self-optimizations, multiple agents

How can we achieve increase?

Increase Scale

generic solutions

SE solutions

SPLE solutions (partitioning of large software projects)

Increase Task

towards reaching productive usage

when and how integrate existing tools

ensure traceability

self-optimizations, multiple agents

Increase Quality (and reliability)

How can we achieve increase?

Increase Scale

generic solutions

SE solutions

SPLE solutions (partitioning of large software projects)

Increase Task

towards reaching productive usage

when and how integrate existing tools

ensure traceability

self-optimizations, multiple agents

Increase Quality (and reliability)

decrease error probability

→ pre-training, and fine-tuning

Increase Scale

generic solutions

SE solutions

SPLE solutions (partitioning of large software projects)

Increase Task

towards reaching productive usage

when and how integrate existing tools

ensure traceability

self-optimizations, multiple agents

Increase Quality (and reliability)

decrease error probability

→ pre-training, and fine-tuning

support existing (hidden) knowledge

→ prompting, model-driven techniques

Increase Scale

generic solutions

SE solutions

SPLE solutions (partitioning of large software projects)

Increase Task

towards reaching productive usage

when and how integrate existing tools

ensure traceability

self-optimizations, multiple agents

Increase Quality (and reliability)

decrease error probability

→ pre-training, and fine-tuning

support existing (hidden) knowledge

→ prompting, model-driven techniques

assessment strategies

→ novel datasets and metrics (real-world projects are too large, accuracy not sufficient)

Increase Scale

generic solutions

SE solutions

SPLE solutions (partitioning of large software projects)

Increase Task

towards reaching productive usage

when and how integrate existing tools

ensure traceability

self-optimizations, multiple agents

Increase Quality (and reliability)

decrease error probability

→ pre-training, and fine-tuning

support existing (hidden) knowledge

→ prompting, model-driven techniques

assessment strategies

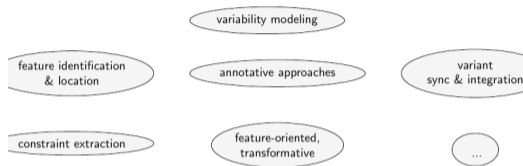
→ novel datasets and metrics (real-world projects are too large, accuracy not sufficient)

increase reliability

→ reasoning for results, certainty values

exercise the levels for one SPL technique

explore combination of classical SPL techniques with Gen-AI support



language-specific, hard to analyze large systems, hard to maintain consistently ...
still insufficient genericity and automation

Business / Tech

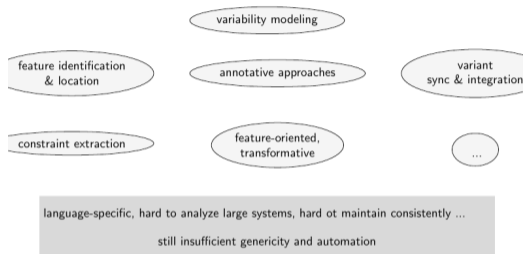
A year after ChatGPT's release, the AI revolution is just beginning



By Catherine Thorbecke, CNN

8 minute read - Updated 10:32 AM EST, Thu November 30, 2023





Business / Tech

A year after ChatGPT's release, the AI revolution is just beginning



By Catherine Thorbecke, CNN

8 minute read - Updated 10:32 AM EST, Thu November 30, 2023



automation	scale	quality
L4 Full Generative AI for Variability	unlimited	expert
L3 Advanced Generative AI for Variability	multiple (projects)	high
L2 Basic Generative AI for Variability	small (project)	low
L1 No Generative AI for Variability	small (files)	low
L0 No Generative AI		

Increase Scale

- generic solutions
- SE solutions
- SPLE solutions (partitioning of large software projects)

Increase Task

- towards reaching productive usage
- when and how integrate existing tools
- ensure traceability
- self-optimizations, multiple agents

Increase Quality (and reliability)

- decrease error probability
→ pre-training, and fine-tuning
- support existing (hidden) knowledge
→ prompting, model-driven techniques
- assessment strategies
→ novel datasets and metrics (real-world projects are too large, accuracy not sufficient)
- increase reliability
→ reasoning for results, certainty values

Thanks!

Feedback, Questions, ...?