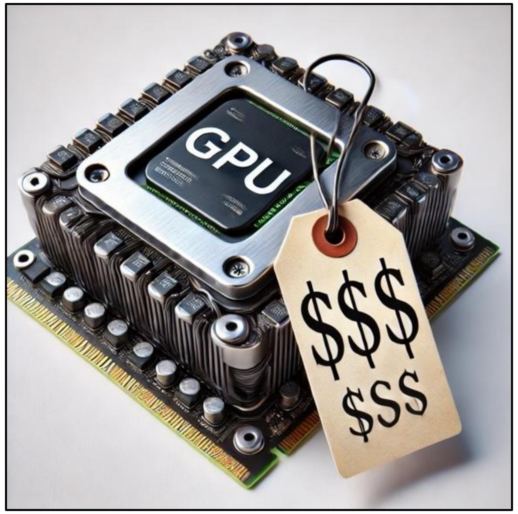
Exploring the Potential of Llama Models in Automated Code Refinement

Genevieve Caumartin, Qiaolin Qin, Sharon Chatragadda, Janmitsinh Panjrolia, Heng Li, Diego Elias Costa

Can Smaller Open-Sourced LMs Measure Up with ChatGPT in Code Refinement Tasks?

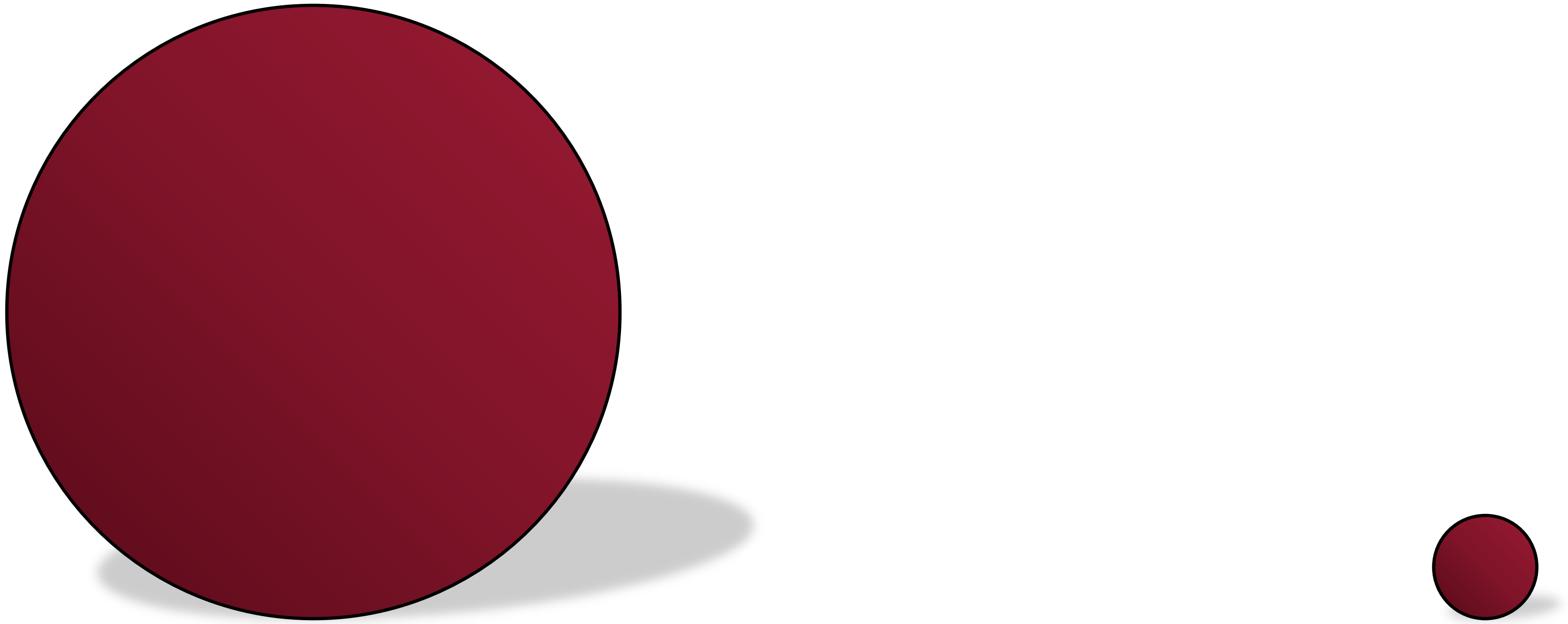
Why Use Smaller, Open-Sourced Models?

- Privacy Concerns
- Recurring Inference Costs
- High-Performance Hardware Costs



Generated by DALL-E

By Smaller we Mean…



ChatGPT3.5
175B parameters

Llama 2 models
7B parameters

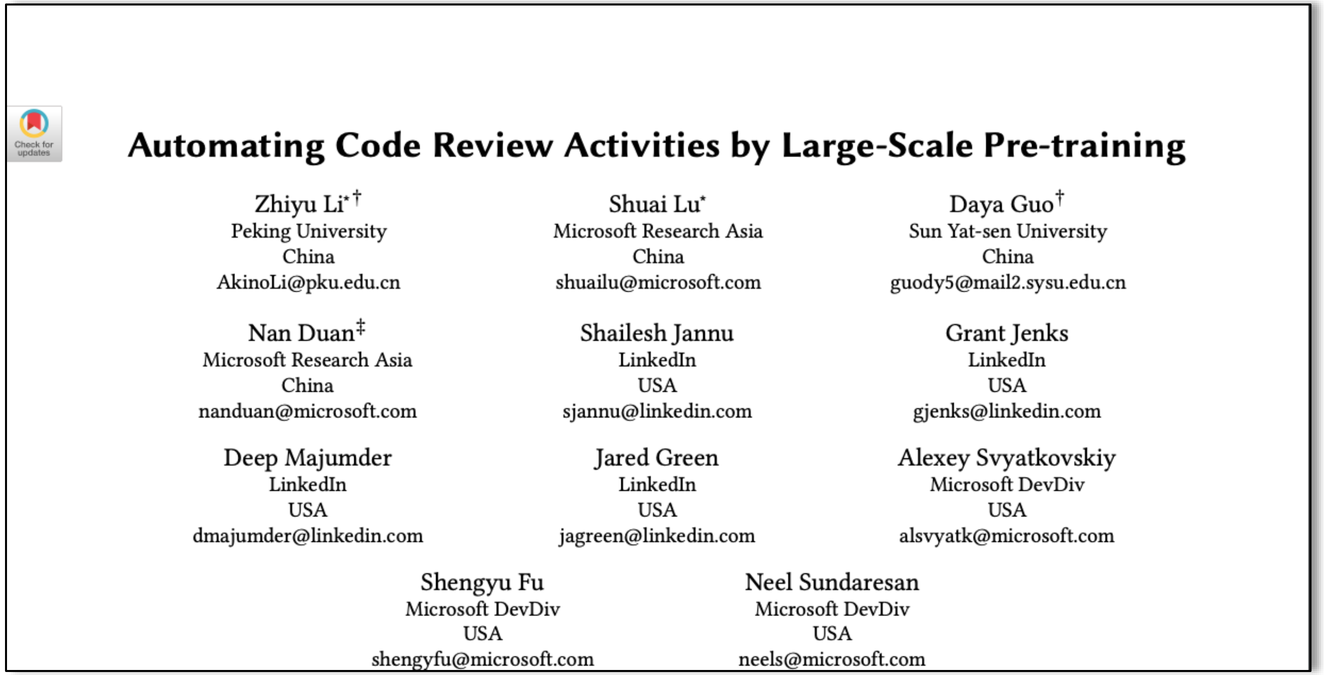
It All Started With

CodeReviewer:

- Pre-trained encoder-decoder
- Trained on code review tasks

ChatGPT3.5:

- General purpose LLM
- One-shot learning on code refinement tasks



Automating Code Review Activities by Large-Scale Pre-training

Zhiyu Li[*†]
Peking University
China
AkinoLi@pku.edu.cn

Shuai Lu[*]
Microsoft Research Asia
China
shuailu@microsoft.com

Daya Guo[†]
Sun Yat-sen University
China
guody5@mail2.sysu.edu.cn

Nan Duan[‡]
Microsoft Research Asia
China
nanduan@microsoft.com

Shailesh Jannu
LinkedIn
USA
sjannu@linkedin.com

Grant Jenks
LinkedIn
USA
gjenks@linkedin.com

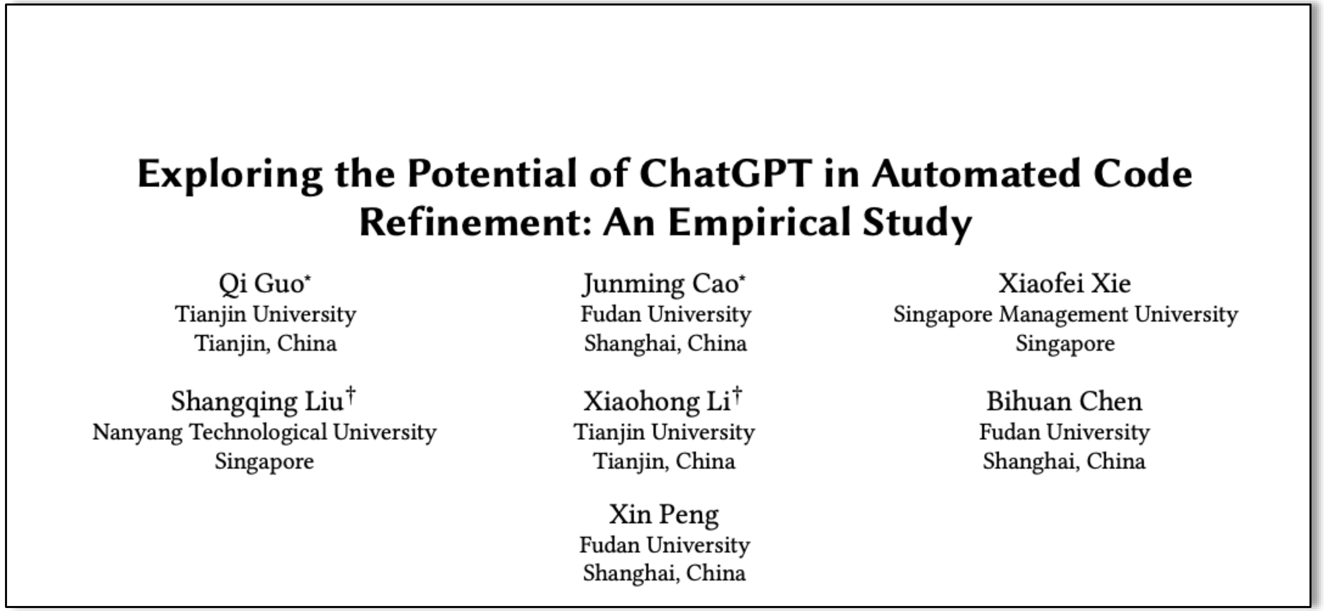
Deep Majumder
LinkedIn
USA
dmajumder@linkedin.com

Jared Green
LinkedIn
USA
jagreen@linkedin.com

Alexey Svyatkovskiy
Microsoft DevDiv
USA
alsvyatk@microsoft.com

Shengyu Fu
Microsoft DevDiv
USA
shengyfu@microsoft.com

Neel Sundaresan
Microsoft DevDiv
USA
neels@microsoft.com



Exploring the Potential of ChatGPT in Automated Code Refinement: An Empirical Study

Qi Guo[*]
Tianjin University
Tianjin, China

Junming Cao[*]
Fudan University
Shanghai, China

Xiaofei Xie
Singapore Management University
Singapore

Shangqing Liu[†]
Nanyang Technological University
Singapore

Xiaohong Li[†]
Tianjin University
Tianjin, China

Bihuan Chen
Fudan University
Shanghai, China

Xin Peng
Fudan University
Shanghai, China

Datasets

CodeReview (CR)
176k code refinement tasks

CodeReview-New (CRN)
15k code refinement tasks

85% train, 7,5% validation, 7,5% test splits

Sample Code Refinement Task

```
pokemon_data = self._get_inventory_pokemon(inventory)
for pokemon in pokemon_data:
    if not(pokemon.get('favorite', 0) is 1 and
        self.config.get('dont_nickname_favorite','')):
```

- Code submitted for review

```
Since `don’t_nickname_favorite` is a Boolean, the `get`
call should default to a Boolean as well (`False`)
```

- Reviewer’s comment

```
pokemon_data = self._get_inventory_pokemon(inventory)
for pokemon in pokemon_data:
    if not(pokemon.get('favorite', 0) is 1 and
      self.config.get('dont_nickname_favorite',False)):
```

- Fix according to comment

Models Under Study

BASELINES

CodeReviewer

ChatGPT3.5 Turbo

TESTED MODELS

Llama 2-Instruct 7B

CodeLlama-Instruct 7B

Evaluation Metrics

Exact Match (EM) / Exact Match-Trim (EM-T)

Evaluates if the code matches the ground truth perfectly

BLEU / BLEU-Trim (BLEU-T)

Calculates 4-gram overlaps

Research Questions

RQ1: Best temperature and prompt settings

RQ2: How do Llama models compare with ChatGPT

RQ3: Factor influencing performance

RQ1: What are the best settings?

Temperatures

- 0, 0.5 and 1.0 temperature settings

✓ Temperature 0 is the best setting for all models

Prompts

- 5 different types of prompts

Each model has its own preference

RQ1: Prompt Building Blocks

Code snippet:``` <code> ```

Code review: <review comment>

Please generate the revised code according to the review […]

Base Prompt

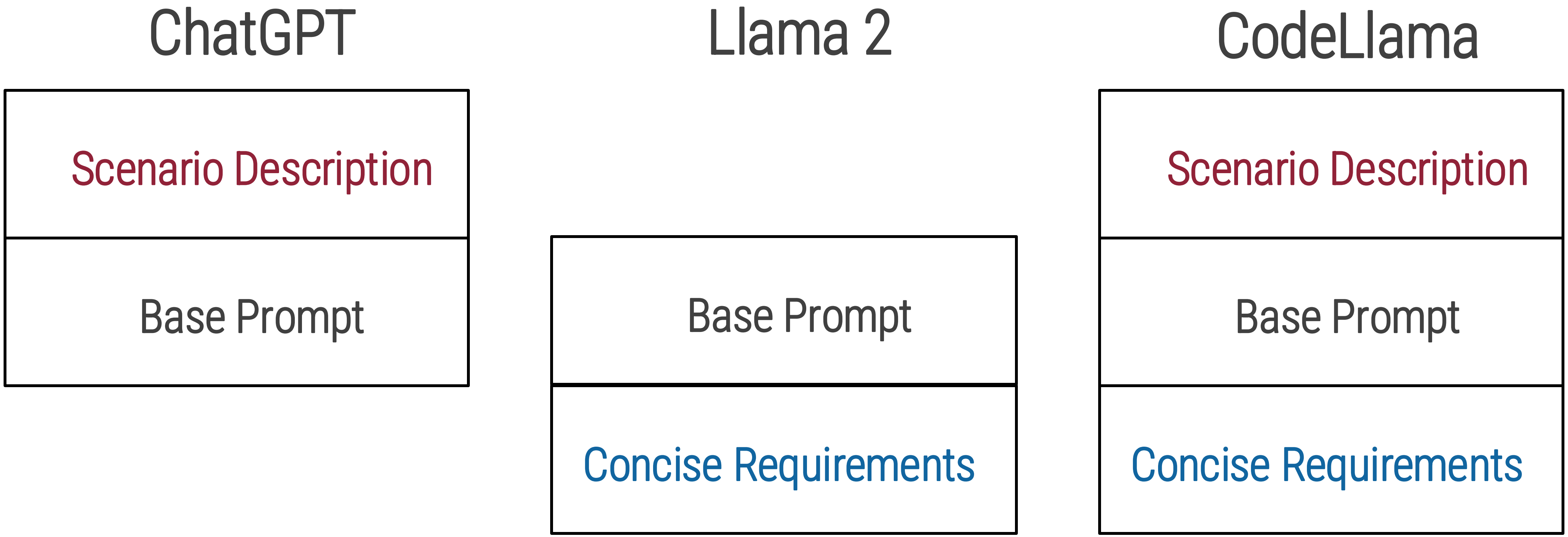
As a developer, imagine you’ve submitted a pull request, and your team leader requests you to make a change in your code […]

Scenario Description

Please generate the revised code according to the review. Ensure that the revised code follows the original code format and comment, unless explicitly required by the review.

Concise Requirements

RQ1: Best Performing Prompts



ChatGPT

Scenario Description

Base Prompt

Llama 2

Base Prompt

Concise Requirements

CodeLlama

Scenario Description

Base Prompt

Concise Requirements

RQ2: How do Llama Models Compare?

- CodeReviewer is #1 on the CR dataset
- CodeLlama beats ChatGPT on BLEU-T
- Llama 2 lags behind

Dataset	Model	EM-T	BLEU-T
CodeReview	CodeReviewer	32.55	83.50
	ChatGPT3.5	19.47	75.12
	CodeLlama	11.89	77.75
	Llama 2	4.98	63.72

RQ2: How do Llama Models Compare?

- CodeReviewer’s performance drops
- CodeLlama is head-to-head with ChatGPT on BLEU-T
- Llama 2 beats CodeReviewer on BLEU-T

Dataset	Model	EM-T	BLEU-T
CodeReview-New	CodeReviewer	15.50	62.88
	ChatGPT3.5	22.78	76.44*
	CodeLlama	13.73	77.13*
	Llama 2	8.56	66.88

* Difference not statistically significant

Not an Exact Match, but Alternate Solution?

- Lower # of exact matches for smaller models
- EM-T is strict; penalizes extra spaces, etc.
- In instances where ChatGPT got an ExactMatch, but not CodeLlama:
  - 48% of CodeLlama’s alternate solutions are valid

APPROVED

RQ3: Factors Influencing Performance

- On 400 tasks categorized by Guo et al. by Comment Information
- Categorizes reviewer’s comment quality

Concrete suggestion:

~~~
```suggestion if not self.available or stability <
self.min_stability: return 0.0 return self.value ```
~~~

Vague question:

Since we're already passing in the DocumentId for the primary document, can we just fetch the linked DocumentIds further down? I'm not sure why we're fetching it here only to pass it through.

Comment Information	CodeLlama		ChatGPT	
	EM-T	**BLEU-T**	**EM-T**	**BLEU-T**
Concrete Suggestion	23.68	84.36	34.74	84.73
Vague Suggestion	1.01	73.60	10.10	73.56
Vague Question	1.80	72.81	6.31	68.21

RQ3: Factors Influencing Performance

- Results for CodeLlama

Type of Change		
	EM-T	BLEU-T
Add Documentation	0.0	47.69
Refactor – Rename	26.47	87.17
Refactor - Conventions	20.83	87.77
Modify Code Logic	15.69	80.82
Documentation and Code	0.0	60.32

- Limited ability for adding documentation
- Better at refactoring and modifying existing code

Latest Llama Model vs CodeLlama

- Llama 3.1-Instruct 8B, improved general-purpose model
- Worse than CodeLlama on EM-T

Dataset	Model	EM-T	BLEU-T
CR	CodeLlama	**11.89**	**77.75**
	Llama 3.1	9.76	75.78
CRN	**CodeLlama**	**13.73**	77.13*
	Llama 3.1	11.59	78.54*

* Difference not statistically significant

Conclusion

- A 25x smaller model shows potential for real-world code review assistance
  - Temp=0 yields best results
  - Data quality important: need concrete suggestions
  - Best at modifying code and refactoring
  - A model fine-tuned on coding tasks is beneficial

Preprint →

✉ GENEVIEVE.CAUMARTIN@MAIL.CONCORDIA.CA

in HTTPS://WWW.LINKEDIN.COM/IN/CBGEN/