

# Large Language Models are Few-Shot Summarizers: Multi-Intent Comment Generation via In-Context Learning

Mingyang Geng  
gengmingyang13@nudt.edu.cn  
College of Computer Science,  
National University of Defense  
Technology  
Changsha, China

Haotian Wang  
wanghaotian13@nudt.edu.cn  
College of Computer Science,  
National University of Defense  
Technology  
Changsha, China

Xiaoguang Mao  
xgmao@nudt.edu.cn  
College of Computer Science,  
National University of Defense  
Technology  
Changsha, China

Shangwen Wang  
wangshangwen13@nudt.edu.cn  
College of Computer Science,  
National University of Defense  
Technology  
Changsha, China

Ge Li  
lige@pku.edu.cn  
Key Lab of High Confidence Software  
Technology, Peking University  
Beijing, China

Xiangke Liao  
xkliao@nudt.edu.cn  
College of Computer Science,  
National University of Defense  
Technology  
Changsha, China

Dezun Dong  
dong@nudt.edu.cn  
College of Computer Science,  
National University of Defense  
Technology  
Changsha, China

Zhi Jin  
zhijin@pku.edu.cn  
Key Lab of High Confidence Software  
Technology, Peking University  
Beijing, China

# 背景

近年、コードコメントを自動生成する技術が盛ん

- プログラムの理解を促進する上で重要

開発者がコードコメントを書くのを忘れたり，時間がなかったするため，プログラム理解を促進する

コードからトークンを抽出するIR技術<sup>[25,58]</sup>やニューラル機械翻訳モデル<sup>[5,28]</sup>などのタスクに適用されている

[5] Uri Alon, Shaked Brody, Omer Levy, and Eran Yahav. 2019. code2seq: Generating Sequences from Structured Representations of Code. In Proceedings of the 7th International Conference on Learning Representations. OpenReview.net.

[25] Sonia Haiduc, Jairo Aponte, Laura Moreno, and Andrian Marcus. 2010. On the use of automated text summarization techniques for summarizing source code. In 2010 17th Working Conference on Reverse Engineering. IEEE, 35–44.

[28] Xing Hu, Ge Li, Xin Xia, David Lo, and Zhi Jin. 2018. Deep code comment generation. In Proceedings of the 26th conference on program comprehension. 200–210.

[58] Paige Rodeghero, Collin McMillan, Paul W McBurney, Nigel Bosch, and Sidney D’Mello. 2014. Improving automated source code summarization via an eyetracking study of programmers. In Proceedings of the 36th international conference on Software engineering. 390–401.

# コードコメント自動生成における問題点

## 開発者の意図を考慮したコメントを生成できていない

- コードとコメントを 1:1 の対応として生成される
- 関連研究
  - コメントには隠された6つのカテゴリを特定<sup>[75]</sup>
  - 約67%のメソッドが複数の意図を含んでいる<sup>[47]</sup>

「1:多」のコメントを生成する必要がある（多意図コメント生成タスク）

## 既存手法

- Mu et al<sup>[47]</sup>はDOMEというアプローチを提案
    - Transformerに基づいた教師あり学習を採用
- **訓練データが限られている**ため、効果が制限される
- データ不足の問題に直面してしまう

[75] Juan Zhai, Xiangzhe Xu, Yu Shi, Guanhong Tao, Minxue Pan, Shiqing Ma, Lei Xu, Weifeng Zhang, Lin Tan, and Xiangyu Zhang. 2020.

CPC: Automatically classifying and propagating natural language comments via program analysis. In Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering. 1359–1371.

[47] Fangwen Mu, Xiao Chen, Lin Shi, Song Wang, and Qing Wang. 2023. DeveloperIntent Driven Code Comment Generation. arXiv preprint arXiv:2302.07055 (2023).

# 多意図コメントタスク解決に向けて

## 大規模言語モデル（LLM）利用を提案

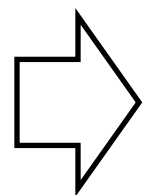
- 動機

- コードと関連するコメントを使用して事前学習されている
- LLMを訓練したプロジェクトのコードコメントには多くの意図がある<sup>[47]</sup>

## インコンテキスト学習を使用

- タスクの詳細な情報を説明する指示
- タスクがうまく行えるかの例
- 解決が求められるクエリ

これを含んだプロンプトを  
モデルに提示



インコンテキスト学習がLLMに効果的な方法である<sup>[8]</sup>

[8] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020.

Language models are few-shot learners. Advances in neural information processing systems 33 (2020), 1877–1901.

# 目的とリサーチクエスチョン

---

## 目的

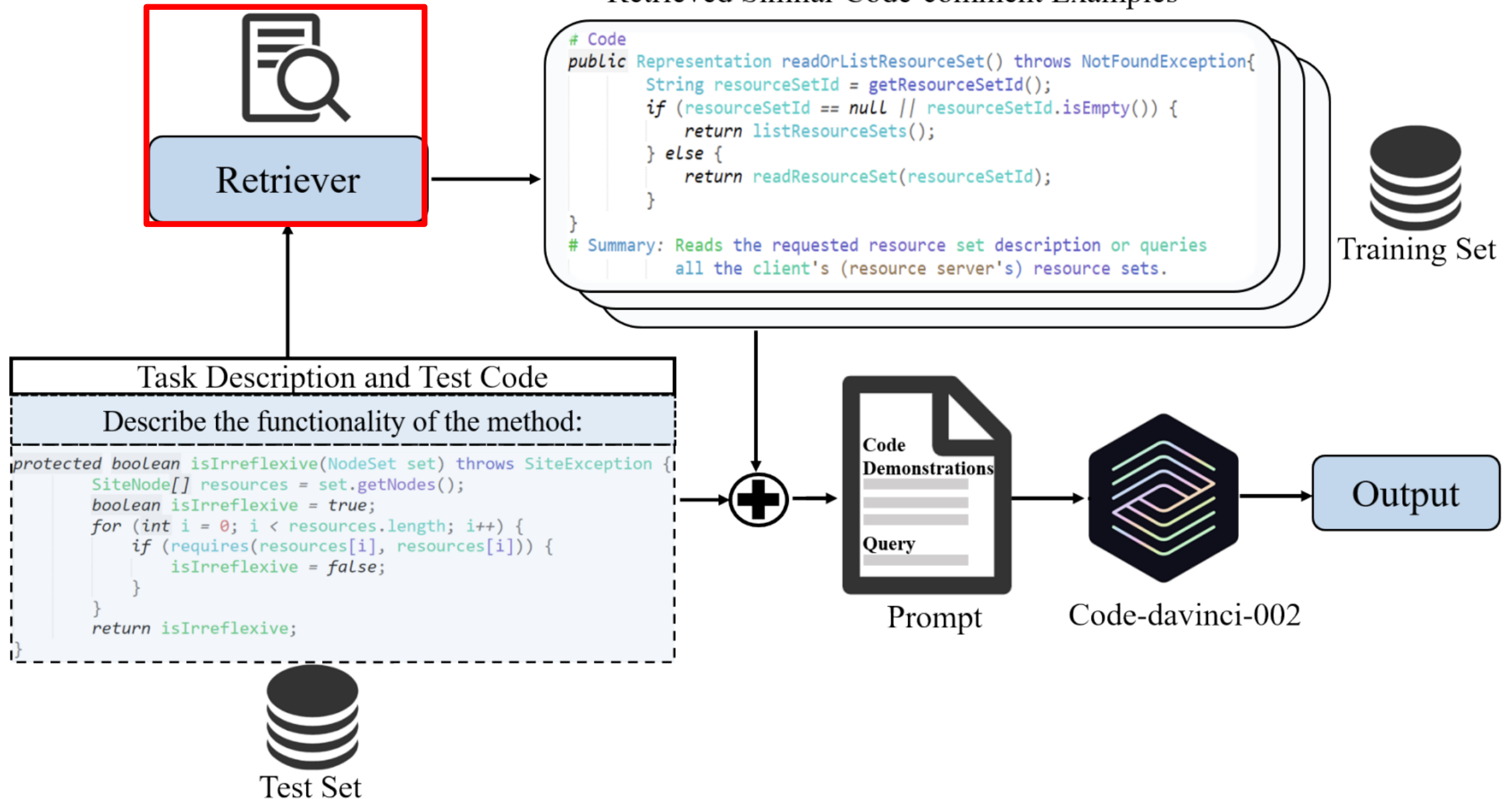
- インコンテキスト学習を使用したLLMが多意図コメント生成において、どの程度有効なのかを明らかにする

## リサーチクエスチョン

- RQ1 : zeroshot, oneshot, fewshotでの有効性は？
- RQ2 : IRベースのデモンストレーション戦略の有効性は？
- RQ3 : 再ランキング戦略での有効性は？

# インコンテキスト学習：概要

RQ1：ランダム  
RQ2：デモンストレーション戦略  
RQ3：再ランキング付け戦略



# 多意図コメント生成のプロンプトテンプレート

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

----- begin of prompt -----

# You are an expert Java programmer, please describe the functionality of the method:  
Natural Language Instruction

# Code

public Representation readOrListResourceSet() throws NotFoundException{  
 String resourceSetId = getResourceSetId();  
 if (resourceSetId == null || resourceSetId.isEmpty()) {  
 return listResourceSets();  
 } else {  
 return readResourceSet(resourceSetId);  
 }  
}

# Summary: Reads the requested resource set description or queries  
all the client's (resource server's) resource sets.

###

Code Demonstration 1

-----

Code Demonstrations

-----

# Code

protected boolean isIrreflexive(NodeSet set) throws SiteException {  
 SiteNode[] resources = set.getNodes();  
 boolean isIrreflexive = true;  
 for (int i = 0; i < resources.length; i++) {  
 if (requires(resources[i], resources[i])) {  
 isIrreflexive = false;  
 }  
 }  
 return isIrreflexive;  
}

# Summary:

----- end of prompt -----

# Groundtruth: Checks if the dependence relation is irreflexive.

自然言語の指示

コードスニペット

意図するコメント

クエリコード

プログラミング言語とコメントの意図

各デモンストレーションごとに、  
コードスニペットと意図するコメント

モデルにクエリコードの  
望ましいコメントを出力させる

7

# RQ2・RQ3：二つの戦略

## デモンストレーション戦略（IRベース）

- トークンベース
  - 類似したコードを識別するのが目的（コードトークンの重複に重点）
  - ターゲットコードとコードスニペットのトークン類似性をランク付け
- セマンティックベース
  - コードのセマンティクスを理解するのが目的
  - $\cos$ 類似度を使用し，コードスニペットに類似したものを探す

## 再ランキング戦略

- ターゲットコードに最も類似しているコードスニペットのコメントを参照し，トークン，セマンティックの二つから類似性を計算

# データセット

既存研究<sup>[47]</sup>で使用されたマルチインテントコメント生成データセットを使用

- Fumcom<sup>[36]</sup>とTLC<sup>[30]</sup>を使用
- 各コメントをカテゴリーでラベル付け

| Dataset        | Funcom    |        | TLC    |       |
|----------------|-----------|--------|--------|-------|
|                | Train     | Test   | Train  | Test  |
| What           | 685,992   | 44,330 | 28,991 | 2,724 |
| Why            | 152,026   | 8,402  | 5,935  | 381   |
| How-to-use     | 24,648    | 1,233  | 838    | 48    |
| How-it-is-done | 146,571   | 6,466  | 11,478 | 687   |
| Property       | 166,459   | 8,326  | 5,016  | 396   |
| Total          | 1,175,696 | 68,757 | 52,258 | 4,236 |

カテゴリー

- 評価に曖昧と見なされるコメントは除外

[36] Alexander LeClair, Siyuan Jiang, and Collin McMillan. 2019. A neural model for generating natural language summaries of program subroutines. In 2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE). IEEE, 795–806.

[30] Xing Hu, Ge Li, Xin Xia, David Lo, Shuai Lu, and Zhi Jin. 2018. Summarizing source code with transferred api knowledge. (2018).

# 実験設定

---

## 評価指標

- BLUE<sup>[51]</sup>, ROUGE-L<sup>[42]</sup>, METEOR<sup>[6]</sup>

## 学習

- 0-shot, 1-shot, 5-shot, 10-shot

## 使用するLLM

- code-davinci-002 (Codexの最新モデル)

[51] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting of the Association for Computational Linguistics. 311–318.

[42] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In Text summarization branches out. 74–81.

[6] Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization. 65–72.

# RQ1

zeroshot, oneshot, fewshotでの有効性は？

| Intent  | Method        | Funcom      |             |             | TLC         |             |             |
|---------|---------------|-------------|-------------|-------------|-------------|-------------|-------------|
|         |               | BLEU        | ROUGE-L     | METEOR      | BLEU        | ROUGE-L     | METEOR      |
| Average | DOME          | 31.8        | 42.5        | 20.5        | 22.2        | 36.7        | 16.5        |
|         | Codex-0-shot  | 21.2        | 25.2        | 12.4        | 18.8        | 21.2        | 11.3        |
|         | Codex-1-shot  | 23.1        | 30.7        | 16.2        | 21.1        | 26.1        | 13.6        |
|         | Codex-5-shot  | 27.4        | 52.9        | 20.6        | 24.4        | 49.9        | 16.8        |
|         | Codex-10-shot | <b>33.4</b> | <b>76.1</b> | <b>24.1</b> | <b>27.2</b> | <b>66.7</b> | <b>19.2</b> |

各カテゴリの平均

0-shot, 1-shotは既存手法と比べて最適でない

- 例) Funcom : zeroで21.2, oneで23.1だがDOMEは31.8

適切にプロンプトされると教師あり学習を超える

- 例) Funcom : 10-shotで33.4, DOMEは31.8

# RQ2 ・ RQ3

IRベースのデモンストレーション戦略の有効性は？  
再ランキング戦略での有効性は？

## RQ2の結果

| Intent  | Method                                                | Funcom      |             |             | TLC         |             |             |
|---------|-------------------------------------------------------|-------------|-------------|-------------|-------------|-------------|-------------|
|         |                                                       | BLEU        | ROUGE-L     | METEOR      | BLEU        | ROUGE-L     | METEOR      |
| Average | Codex-1-shot                                          | 23.1        | 30.7        | 16.2        | 21.1        | 26.1        | 13.6        |
|         | Codex-1-shot ( <i>Selection<sub>token</sub></i> )     | <b>39.2</b> | <b>77.3</b> | <b>35.1</b> | <b>36.1</b> | <b>72.8</b> | 31.0        |
|         | Codex-1-shot ( <i>Selection<sub>semantic</sub></i> )  | 37.8        | 73.9        | 35.0        | 35.1        | 68.3        | <b>31.9</b> |
|         | Codex-5-shot                                          | 27.4        | 52.9        | 20.6        | 24.4        | 49.9        | 16.8        |
|         | Codex-5-shot ( <i>Selection<sub>token</sub></i> )     | 41.1        | <b>85.7</b> | 38.8        | 37.6        | <b>79.9</b> | 34.4        |
|         | Codex-5-shot ( <i>Selection<sub>semantic</sub></i> )  | <b>44.9</b> | 82.8        | <b>40.6</b> | <b>41.7</b> | 79.7        | <b>38.0</b> |
|         | Codex-10-shot                                         | 33.4        | 76.1        | 24.1        | 27.2        | 66.7        | 19.2        |
|         | Codex-10-shot ( <i>Selection<sub>token</sub></i> )    | 64.5        | 88.0        | 63.9        | 60.7        | 82.6        | 59.5        |
|         | Codex-10-shot ( <i>Selection<sub>semantic</sub></i> ) | <b>65.9</b> | <b>89.4</b> | <b>65.8</b> | <b>62.8</b> | <b>83.2</b> | <b>62.7</b> |

## RQ3の結果

| Intent  | Method                                                                                   | Funcom      |             |             | TLC         |             |             |
|---------|------------------------------------------------------------------------------------------|-------------|-------------|-------------|-------------|-------------|-------------|
|         |                                                                                          | BLEU        | ROUGE-L     | METEOR      | BLEU        | ROUGE-L     | METEOR      |
| Average | Codex-1-shot                                                                             | 23.1        | 30.7        | 16.2        | 21.1        | 26.1        | 13.6        |
|         | Codex-1-shot ( <i>rerank<sub>token</sub></i> )                                           | 29.1        | 68.9        | 25.2        | 26.8        | <b>64.0</b> | <b>23.2</b> |
|         | Codex-1-shot ( <i>rerank<sub>semantic</sub></i> )                                        | <b>30.2</b> | <b>69.0</b> | <b>25.8</b> | <b>27.2</b> | 63.9        | 22.2        |
|         | Codex-1-shot ( <i>Selection<sub>token</sub></i> )                                        | 39.2        | 77.3        | 35.1        | 36.1        | 72.8        | 31.0        |
|         | Codex-1-shot ( <i>Selection<sub>token</sub></i> + <i>Rerank<sub>token</sub></i> )        | <b>44.0</b> | <b>82.0</b> | <b>41.9</b> | <b>41.5</b> | <b>77.4</b> | <b>37.9</b> |
|         | Codex-1-shot ( <i>Selection<sub>token</sub></i> + <i>Rerank<sub>semantic</sub></i> )     | 43.7        | 80.8        | 40.9        | 41.4        | 76.4        | 37.7        |
|         | Codex-10-shot                                                                            | 33.4        | 76.1        | 24.1        | 27.2        | 66.7        | 19.2        |
|         | Codex-10-shot ( <i>rerank<sub>token</sub></i> )                                          | 35.1        | <b>88.3</b> | <b>27.4</b> | 32.0        | <b>83.8</b> | <b>24.5</b> |
|         | Codex-10-shot ( <i>rerank<sub>semantic</sub></i> )                                       | <b>35.3</b> | 87.1        | 27.4        | <b>32.4</b> | 83.8        | 24.2        |
|         | Codex-10-shot ( <i>Selection<sub>semantic</sub></i> )                                    | 65.9        | 89.4        | 65.8        | 62.8        | 83.2        | 62.7        |
|         | Codex-10-shot ( <i>Selection<sub>semantic</sub></i> + <i>Rerank<sub>token</sub></i> )    | <b>72.4</b> | <b>91.8</b> | <b>71.6</b> | <b>68.8</b> | <b>86.3</b> | 68.6        |
|         | Codex-10-shot ( <i>Selection<sub>semantic</sub></i> + <i>Rerank<sub>semantic</sub></i> ) | 72.0        | 91.6        | 71.5        | 68.4        | 85.7        | <b>68.9</b> |

## 結果

2つの戦略はCodexの有効性を向上させた

# 人間評価

## BLUEなどの指標は語彙的な違いを評価する

- 意味的な違いは不十分

→ コメントの質をさらに評価するため、人間評価を実施

## 実験設定

- 参加者：5年以上のJava開発経験者6名
- 100個のコードスニペットをランダムに選択
  - 各カテゴリーから20個ずつ

DOME

Codex-10-shot

Codex-10-shot(semantic)

Codex-10-shot(semantic & token)

それぞれのモデルごとに100個選択  
→ 合計400の生成コメントを用意

# 人間評価：結果

## 3つの側面から評価

1. 文法の観点から自然なコメントが生成されているか
2. 生成されたコメントの情報を豊かさ
3. 生成されたコメントが開発者にどの程度役立つか

→ 5段階のリッカー尺度（1：悪い～5：優れている）で評価

|             | Approach                                                | Avg.       | Std. |
|-------------|---------------------------------------------------------|------------|------|
| Naturalness | DOME                                                    | 3.9        | 0.8  |
|             | Codex-10-shot                                           | 4.2        | 0.7  |
|             | Codex-10-shot ( $Selection_{semantic}$ )                | 4.3        | 0.8  |
|             | Codex-10-shot ( $Selection_{semantic}+Rerank_{token}$ ) | <b>4.3</b> | 0.7  |
| Adequacy    | DOME                                                    | 3.3        | 1.3  |
|             | Codex-10-shot                                           | 3.5        | 1.1  |
|             | Codex-10-shot ( $Selection_{semantic}$ )                | 3.8        | 1.2  |
|             | Codex-10-shot ( $Selection_{semantic}+Rerank_{token}$ ) | <b>4.1</b> | 0.9  |
| Usefulness  | DOME                                                    | 3.0        | 1.4  |
|             | Codex-10-shot                                           | 3.1        | 1.3  |
|             | Codex-10-shot ( $Selection_{semantic}$ )                | 3.7        | 1.1  |
|             | Codex-10-shot ( $Selection_{semantic}+Rerank_{token}$ ) | <b>3.8</b> | 1.3  |

LLMは自然言語の説明を生成するのが得意であることがわかる

# まとめ

---

インコンテキスト学習を使用したLLMが多意図コメント生成での有効性を評価

5-shot, 10-shotで教師あり学習（DOME）を超えた

- 適切にプロンプトを与える必要がある

IRベースのデモンストレーション戦略, 再ランキング戦略の共にLLMの有効性を向上させた