# Can LLMs Replace Manual Annotation of Software Engineering Artifacts?

Toufique Ahmed, Premkumar Devanbu, Christoph Treude, Michael Pradel, 4 Feb 2025.

MSR 2025 Technical Papers

# 背景

## SEツールの評価

ソフトウェア工学のツールやプロセスは人間による評価が重要.

例：自動生成コードの要約，静的解析ツールの警告の修正など

## 評価における課題

人間評価は専門家を呼ぶ必要があるため，高コストで時間がかかる.

➤LLMを使用してこの評価コストを削減できないか？

# 研究概要

## ４つのリサーチクエスチョン

RQ1：人間とモデルの，**評価の一致度**はどの程度か？

RQ2：あるタスクが**LLMに不向きかどうか**をどう判断するか？

RQ3：LLMの回答が信頼できる，**特定のデータ**をどう見分けるか？

RQ4：どれくらいの**人的労力を削減**できるか？

## アプローチ手法

過去の研究で人間が行ったアノテーションタスクを，LLMに実施させ，その結果を人間評価と比較・分析を行う.

# 研究概要

## 対象タスクとデータセット

5つのデータセットと10種類のアノテーションタスクを対象とした.

### コード要約の品質評価

・生成されたコード要約の**正確性，妥当性，簡潔性，類似性を評価**.

### 名前と値の不一致

・変数名とその中身が**一致しているかを評価**.

### 因果関係の検出

・要求仕様書から**因果関係を抽出**.

### 意味的類似性

・2つのJavaメソッドが機能的に**似ているかを評価（目的，操作，効果）**.

### 静的解析警告の判断

・コード修正が静的解析ツールの警告を**解決したかを判断**.

# 研究概要

## 使用したLLMモデル

クローズドモデル

・GPT-4， GPT-3.5， Claude-3.5-Sonnet， Gemini-1.5-Pro

オープンモデル

・Llama3(70B)， Mixtral(8x22B)

## 評価指標

評価者間の一致度を測る「クリッペンドルフのα係数」を使用.

$$\alpha = 1 - \frac{D_o}{D_e}$$

$D_o$ : 観測された不一致度

$D_e$ : 期待される不一致度

# 結果 1– 人間とモデルの一致度

## クリッペンドルフのα係数で比較

各タスクで人間とLLMの一致度を比較

| タスクのカテゴリ | 人間同士 | 人間とモデル | モデル同士 |
|---|---|---|---|
| コード要約 (正確性) | 0.44 | 0.48 | 0.78 |
| 名前と値の不一致 | 0.52 | 0.48 | 0.66 |
| 意味的類似性 (目標) | 0.83 | 0.77 | 0.81 |
| 静的解析警告 | **0.80** | **0.15** | **0.12** |
| 因果関係 | **0.49** | **0.28** | **0.25** |

ほとんどのタスクで人間とモデルは人間同士と同等の数値であった.

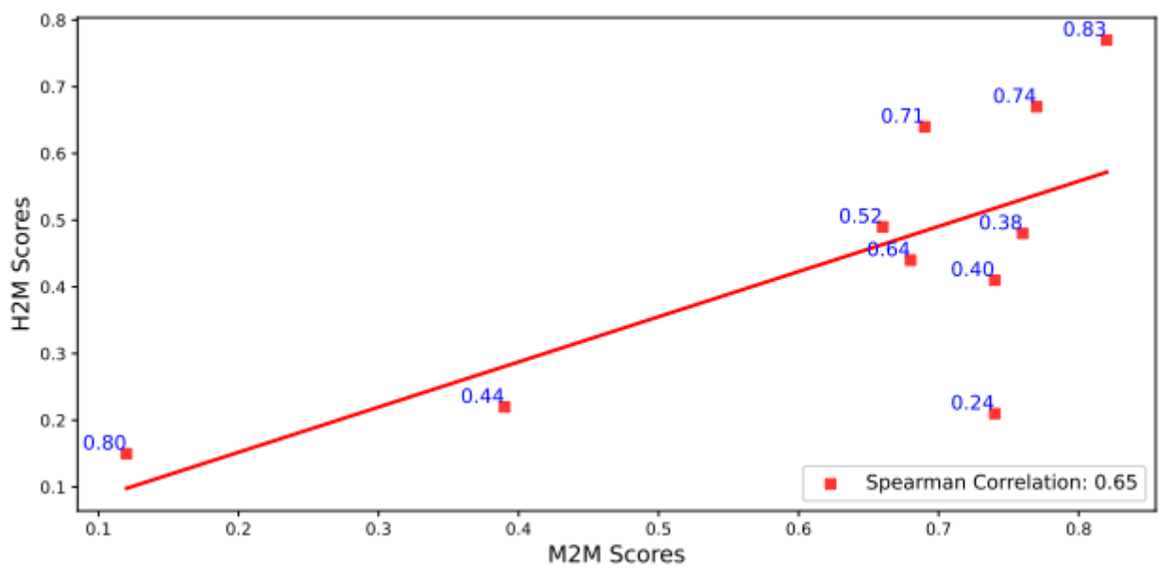
例外：静的解析警告の判断，因果関係の検出

➤モデルは文章の論理的判断，複数の情報の統合が不得意と考えられる.

# 結果 2– LLMのタスクへの適応度合

## スピアマン相関係数を使用して相関関係を確認.

複数のLLMに同タスクを行わせLLM同士の一致度と，人間とLLMの一致度を比較.




縦軸：人間とモデルの一致度
横軸：モデル同士の一致度

２つの一致度には正の相関があることが確認された.

➤LLM同士の一致度を測るだけで，そのタスクでLLMが人間とどの程度一致しそうかを予測できる.

# 結果 3– 信頼できるデータの見分け方

## LLMの回答の信頼度が指標として利用できるかを検証.

1，あるデータの評価者の1人をLLMに置換する.

①信頼度に基づいてデータを選択

②ランダムにデータを選択　　の２つの手法でLLMに置換.

2，一致度の変化を観察

LLMに置換するデータの割合を徐々に増やしていき，α係数の変化を観察.

## 結果

①の方が，どのデータにおいても一致度がある程度は保たれた.

➤LLMの回答の信頼度は正しさの指標として良いものである.

# 結果 4– 人的労力の削減

## 削減できる２つの要因を確認

①各データに必要な評価者数：1人だけをLLMに置換.

②LLMに置換できるデータの割合：RQ3の結果に基づいて割合を確認.

| Datasets | #of Rating for Each Sample | % Effort Saved for One Rating | % Effort Saved for Overall Process |
|---|---|---|---|
| Accuracy (Code Summarization) | 3 | 100% | 33% |
| Adequacy (Code Summarization) | 3 | 100% | 33% |
| Conciseness (Code Summarization) | 3 | 100% | 33% |
| Similarity (Code Summarization) | 3 | 50% | 16.5% |
| Nalin | 11 | 100% | 9% |
| Causality | 2 | 60% | 30% |
| Goals (Sesame) | 3 | 100% | 33% |
| Operations (Sesame) | 3 | 100% | 33% |
| Effects (Sesame) | 3 | 100% | 33% |
| Static Analysis Warning | 2 | 50% | 25% |

10タスク中7タスクで，人間1人をLLMに置換しても問題なかった.

➤LLMによって人間の作業量を大幅に削減することが可能.

# 理想のLLM導入手順

## LLMを評価に導入するための実践的な手順

１，3,4個のサンプルを人間が作成する.

２，複数のLLMに全サンプルを評価させる.

３，LLM同士の一致度を計算する.

４，一致度によって手順を変更

**①一致度が高い時($\alpha > 0.5$)**

・評価者1人をLLMに安全に置き換えれる.

**②一致度が低い時($\alpha \leq 0.5$)**

・LLMの出力確率が高いデータに限定して選択的に置換する.

# まとめ

## LLMはコストと労力を削減できる有望なツール

ソフトウェア工学においてLLMは有望なツールであった.

- モデル同士の一致度でタスクの向き不向きを判断可能.
- モデルの確信度で任せるべきサンプルを選べる.

## 今後の展望

- 人間の思考プロセスも収集し．プロンプトの改善に繋げる.
- カテゴリ分類だけでなく自由な質的コーディングへの応用も探る.