

From LLMs to Agents in Programming: The Impact of Providing an LLM with a Compiler

大平研究室

発表者: s2620113 福井遥陽

背景

近年の大規模言語モデルの活躍

- 自然言語処理
- プログラム生成
- ソフトウェア開発
 - 依然として、品質要件を満たせてない場合も散見



課題解決

トレーニングデータの増強, モデルサイズの向上

背景

近年の大規模言語モデルの活躍

- 自然言語処理
- プログラム生成
- ソフトウェア開発
 - 依然として、品質要件を満たせてない場合も散見

リソースや労力が必要

トレーニングデータの増強, モデルサイズの向上



性能向上するが, 「品質保証の欠如」が未解決

背景

LLMエージェントの登場

- 生成と「解法の検証能力」を組み合わせたプログラム生成システム
 - コンパイラなどの外部ツールをエージェントと統合することで協調とフィードバックを通じて品質向上させている

既存の研究でLLMエージェントに着目した研究がされている

背景

LLMエージェントの登場

- 生成と「解法の検証能力」を組み合わせたプログラム生成システム
 - コンパイラなどの外部ツールをエージェントと統合することで協調とフィードバックを通じて品質向上させている

既存の研究でLLMエージェントに着目した研究がされている

→しかし, 「LLM+コンパイラ」という最小限の構成においてモデルのサイズやエラーの種類ごとにどんな効果があるのかが未検証

本研究の目的

単一モデルエージェントとコンパイラとの相互作用の調査

リサーチクエスチョン

- RQ1
- RQ2
- RQ3

本研究の目的

リサーチクエスチョン

- RQ1
 - モデル単体に対する，エージェントシステムが生成したプログラムのコンパイル成功率の向上率
- RQ2
 - エージェントシステムの恩恵を最も受けるタスクにはどのようなパターンがあるか
- RQ3
 - エージェントシステムの恩恵を受けるコンパイルエラーの種類にはどのようなパターンがあるか

本研究の目的

リサーチクエスチョン

- RQ1
 - モデル単体に対する，エージェントシステムが生成したプログラムのコンパイル成功率の向上率
- RQ2
 - エージェントシステムの恩恵を最も受けるタスクにはどのようなパターンがあるか
- RQ3
 - エージェントシステムの恩恵を受けるコンパイルエラーの種類にはどのようなパターンがあるか

本研究の目的

リサーチクエスチョン

- RQ1
 - モデル単体に対する，エージェントシステムが生成したプログラムのコンパイル成功率の向上率
- RQ2
 - エージェントシステムの恩恵を最も受けるタスクにはどのようなパターンがあるか
- RQ3
 - エージェントシステムの恩恵を受けるコンパイルエラーの種類にはどのようなパターンがあるか

研究手法

使用データセット

- RosettaCode
 - C言語を対象都市, コンパイルが成功する699個のタスクをサブセットとして抽出

ベースライン

- 16種類の言語モデルそれぞれに対し, プロンプトを与え一回の試行でコード生成させ類似度とコンパイル可否を評価

研究手法

エージェントフレームワーク

1. タスク説明からC言語コードを生成
2. コンパイル成功：処理終了
3. コンパイルエラー：エラーメッセージと前回のコードを集約しフィードバック
4. 最大5回まで反復

研究手法

分析方法

- コンパイル成功率
- テキスト類似度
 - BLEUおよびROUGE-1
- 意味的類似度
 - CodeBERTScore
- タスク・エラー分析
 - タスク説明や正解コードのトークン数との相関
 - 失敗出力の6種類のエラー分類

実験結果

RQ1：コンパイラアクセスによる実行可能プログラム生成率の向上

モデル	サイズ	ベースライン	エージェント	改善幅
Qwen3	4B	18.0%	97.9%	+79.4%
Code Llama	7B	48.2%	78.8%	+30.6%
Phi-4	14B	84.7%	98.3%	+13.6%
Llama 3.3	70B	92.0%	99.9%	+7.9%
SmolLM 2	360M	7.4%	25.5%	+18.1%

※コンパル成功率

- Qwen 3(4B)は全てのベースラインを上回った.
- ほぼ全てのモデルで、正解したプログラムが反復回数3回以内に生成された

実験結果

RQ2：最も恩恵を受けるタスクのパターン

分析対象	指標/相関係数（ r ）
タスクカテゴリ	63カテゴリの分布比較
正解コード長	$r = 0.118$ (トークン数)
問題文の長さ	$r = 0.127$ (トークン数)

- タスク種類に依存せず，幅広いタスクで一律にコンパイラのフィードバックが有効に機能した

実験結果

RQ3：最も恩恵を受けるエラーのパターン

エラー分類	エラー発生数	削減率	要因・特徴
未定義参照	121件→16件	-86.8%	Main関数や関数定義漏れなど
構文エラー	351件→86件	-75.5%	エラー行と原因が明確
言語不一致	179件→101件	-43.6%	エラー文から原因を誤解しやすい
マークダウンエラー	84件→54件	-35.7%	コード抽出失敗など

- エラー箇所と原因が明確なエラーは70%以上削減
- 指示無視・フォーマット揺れ等の曖昧エラーは改善率が50%を下回った

考察

妥当性の脅威

- 外的妥当性
 - 単一データセット且つC言語に限定されているため，他のプログラムに対しても一般化できるかは不明
- 内的妥当性
 - 静的コンパイル・類似度評価である点
 - ベースラインが一回試行なのに対して，エージェントが最大5回の試行を行っている点

結論

- エージェントシステムの利用により，コンパイル成功率が5.3~79.4%の向上
- Qwen 3のように，小型のエージェントシステムが大規模モデルのベースラインを凌駕

今後の課題

- まず小型モデルで実行し，必要に応じて大型モデルにフォールバックする「階層型モデルシステム」の構築