

Beyond Final Code: A Process-Oriented Error Analysis of Software Development Agents in Real- World GitHub Scenarios

Zhi Chen, Wei Ma, Lingxiao Jiang
Singapore Management University

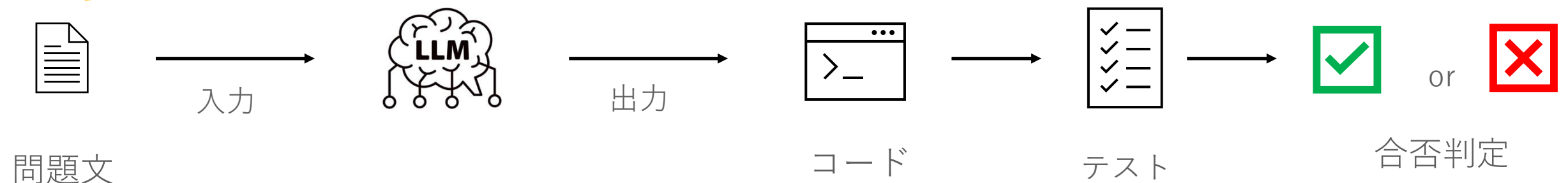
The 48th IEEE/ACM International Conference on Software Engineering (ICSE 2026)

背景

従来のコード生成のベンチマーク

- 関数レベルの問題（EvalPlus, BigCodeBenchなど）

与えられた整数が素数かどうか判定する関数を実装しなさい



- 問題点

関数レベルでは現実のSEの複雑さを十分に捉えられていない



より複雑なベンチマークの作成

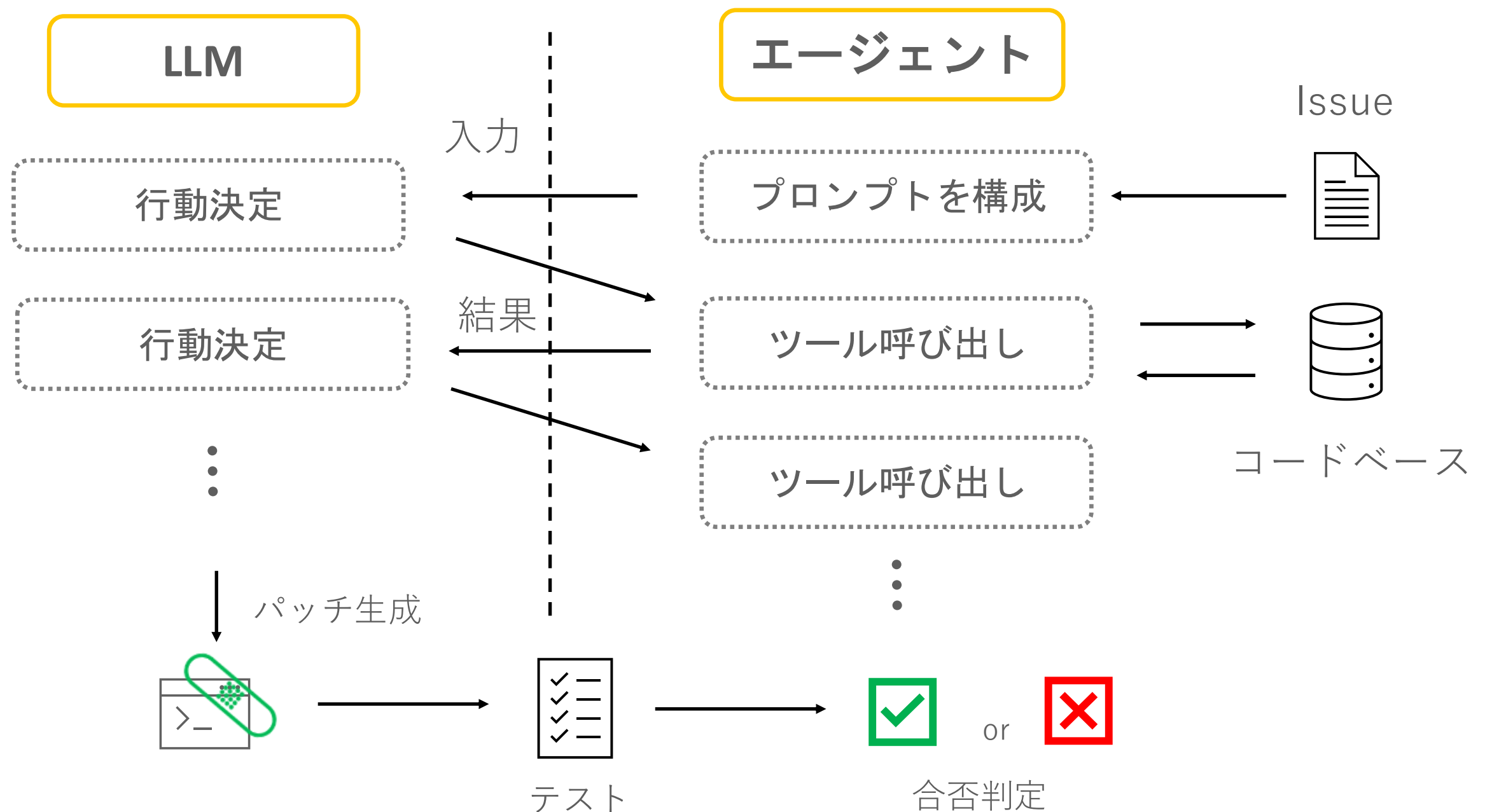
•

エージェント型のアプローチが登場

より複雑なベンチマーク

リポジトリレベルの課題を解決させる

- GitHubのIssueを解決させるベンチマーク



既存研究の問題点

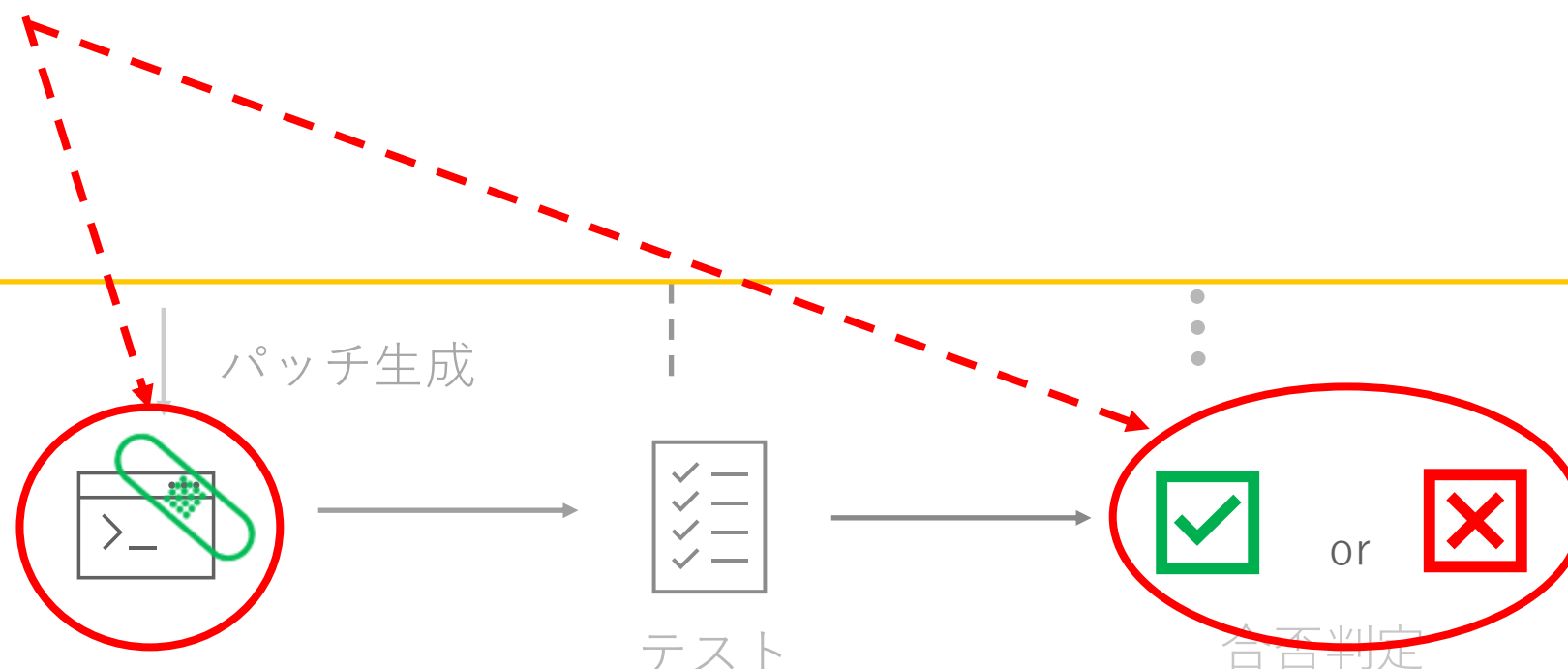
リポジトリレベルの課題を解決させる

- GitHubのIssueを解決させるベンチマーク

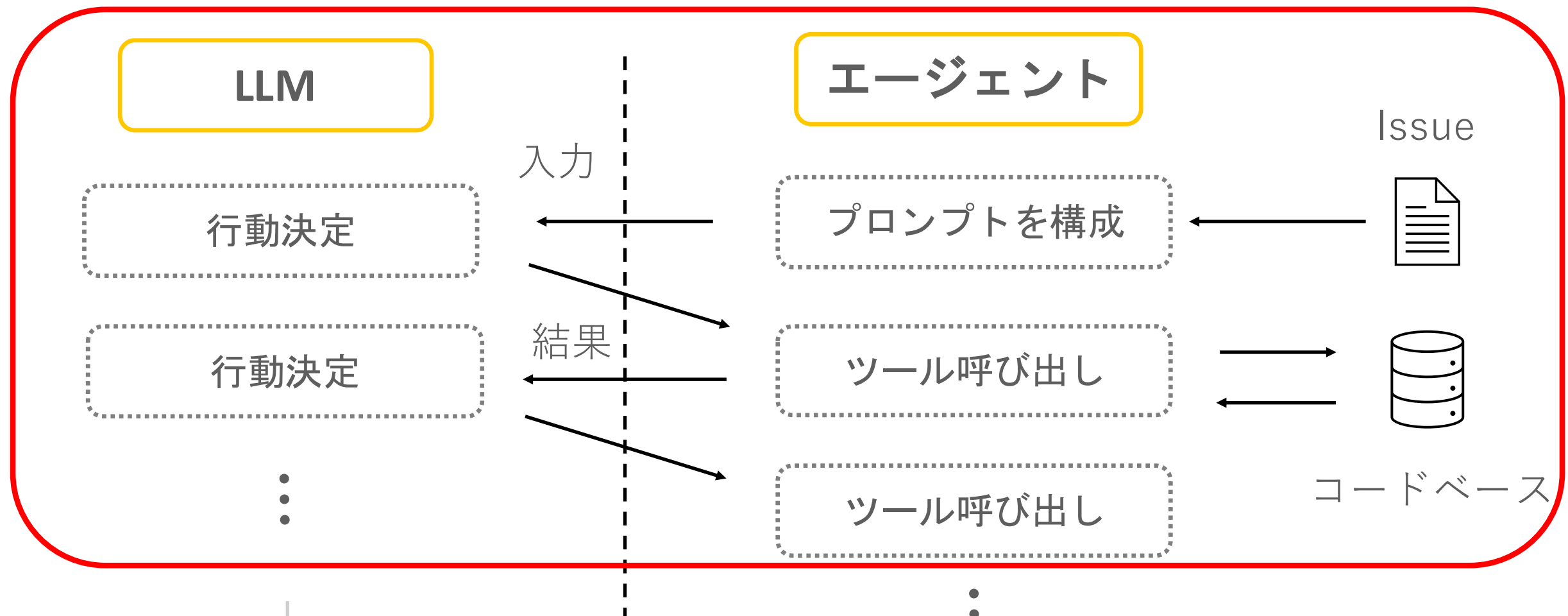
・ 問題点

既存研究はパッチの静的解析 or テスト結果を見ている

最終結果しか分析対象にしていない



既存研究の問題点



- ・ 反復的・動的なプロセスを経ている
 - ・ 同じ結果でも、プロセスは違う可能性
- **解決過程でどのような問題が起きているかを見る必要がある**

目的・アプローチ

目的

- エージェントがIssueを解く過程で発生するエラーを分析して、現在のエージェントの能力と限界を明らかにする

アプローチ

- プロセスデータを集めて分析する
- RQs
 - 解決過程での失敗は最終的な品質にどのように影響するか
 - よく遭遇するエラーは何か
 - 回復するのに特に苦戦するエラーはどのようなタイプか
 - 最終提出でも解決されなかったエラーはどのようなタイプか

調査対象

SWE-bench Verifiedの公開されている実行記録

8つのエージェントの実行記録を利用

- ・ 軌跡ファイル（エージェントの作業記録）
- ・ テストログ（採点結果）
- ・ メタデータ（使ったモデル、解決率など）が含まれる

モデル	解決率	ベースLLM
W&B Programmer	64.6	o1
Blackbox AI Agent	62.8	Unknown
Devlo	58.2	Unknown
Code Act	53.0	Claude 3.5 Sonnet
Engine Labs	51.8	Claude 3.5 Sonnet
Bytedance MarsCode Agent	50.0	Unknown
Tools	49.0	Claude 3.5 Sonnet
CodeShell	44.2	Gemini-2.0-flash-exp

調査結果 | RQ1 (品質への影響)

過程で出力されたコードにエラーがあるかどうか

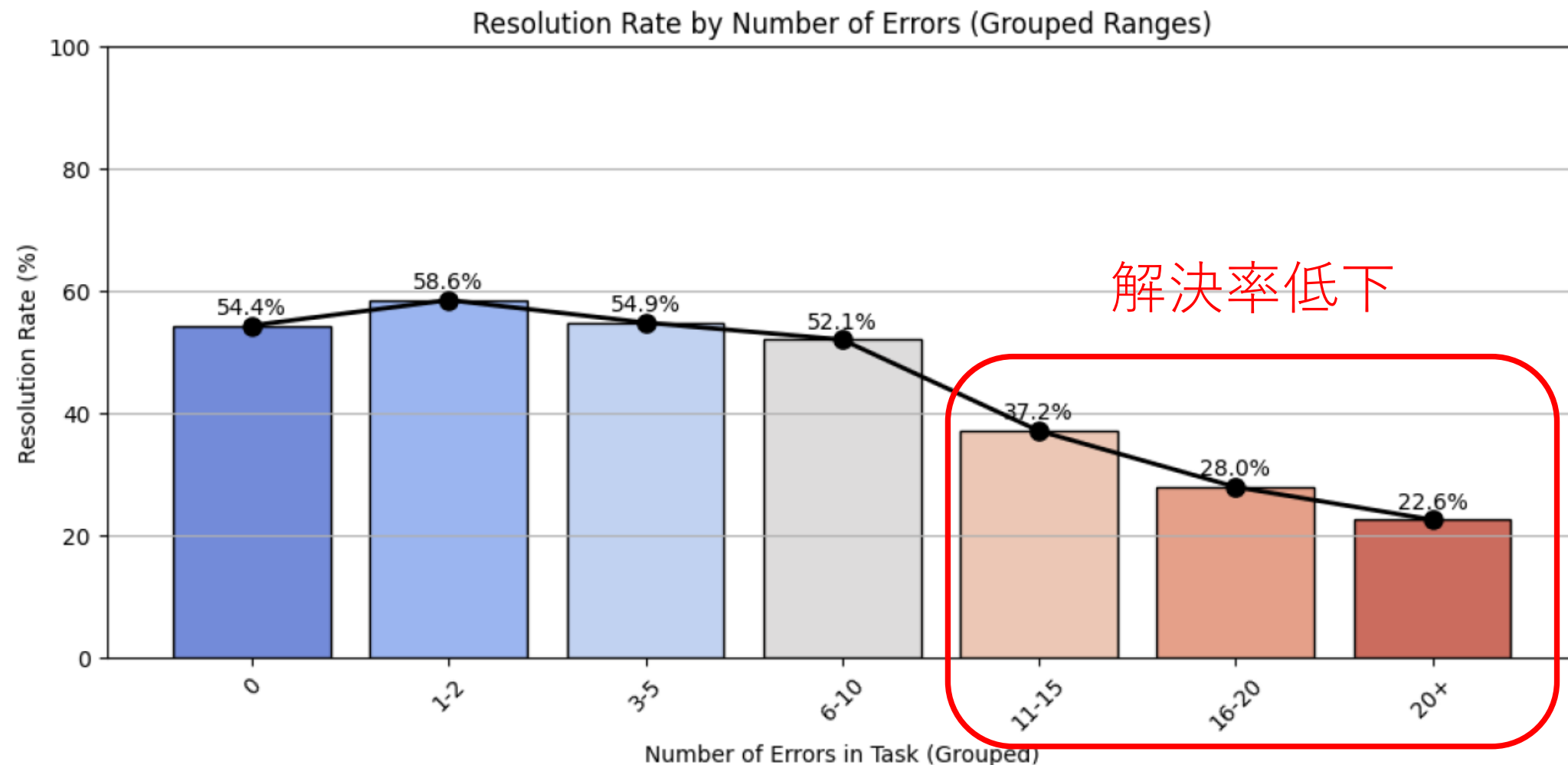
モデル	エラーあり	エラーなし
W&B Programmer	63.55%	64.89%
Blackbox AI Agent	61.49%	72.51%
Devlo	56.85%	60.98%
Code Act	52.72%	53.40%
Engine Labs	54.33%	48.24%
MarsCode	55.81%	45.61%
Tools	49.63%	48.70%
CodeShell	41.03%	45.64%
全体	54.61%	大差なし 54.42%

エージェントによっては差があるが全体ではほとんど差がない

エラーが「起きたこと自体」は最終結果に影響しない

調査結果 | RQ1 (品質への影響)

エラー回数の影響は？



数回のエラーなら問題ないが、**10回を超えると急激に解決率が低下**
エラー頻度は**推論ステップの増加**とも相関していた（コスト増）

調査結果 | RQ2 (よく出るエラーは?)

カテゴリ I : Python組み込みエラー

- Python開発でよく起こる汎用的なエラー

サブカテゴリ	代表例	頻度
依存関係	ModuleNotFoundError	1053回 (最多)
	ImportError	462回
解析・構文	SyntaxError	163回
	IndentationError	36回
アクセス	TypeError	992回 (2位)
	AttributeError	841回 (3位)
システム	OSError	32回
	FileNotFoundError	88回

調査結果 | RQ2 (よく出るエラーは?)

カテゴリ I : Python組み込みエラー

- Python開発でよく起こる汎用的なエラー

サブカテゴリ	何の弱さか?	代表例
依存関係	外部環境の構成	ModuleNotFoundError
解析・構文	生成コードの自己検証	SyntaxError
アクセス	型・変数の追跡	TypeError
システム	OSレベル操作	OSError

つまづきやすい

調査結果 | RQ2 (よく出るエラーは?)

カテゴリⅡ: カスタム定義例外

- 特定のフレームワークやライブラリが独自に定義した例外

サブカテゴリ	代表例	頻度
データベース	<code>django.db.utils.OperationalError</code>	528回
フレームワーク検証	<code>FieldError</code>	107回
管理コマンド	<code>SystemCheckError</code>	30回
ツール/拡張	<code>sphinx.errors.ExtensionError</code>	6回

調査結果 | RQ2 (よく出るエラーは?)

カテゴリⅡ: カスタム定義例外

- 特定のフレームワークやライブラリが独自に定義した例外

サブカテゴリ

代表例

頻度

データベース

`django.db.utils.OperationalError`

528回

カテゴリⅡ (全 1,633件) 内で

約85%がデータベース関連 (8種)

データベース関連の知識が不足している

調査結果 | RQ3 (解決しづらいエラーは?)

再発率 (RR)を計算

- 同じタスク内で同じエラーが繰り返す確率

15種類は再発しなかった (全32種中)

再発率上位4項目を抜粋

エラータイプ	再発率
OSError	71.43%
django.db.utils.IntegrityError	57.69%
IndentationError	50.00%
sqlite3.IntegrityError	50.00%

頻繁に現れた依存関係関連は簡単に修正されている

調査結果 | RQ3 (解決しづらいエラーは?)

再発率 (RR)を計算

- 同じタスク内で同じエラーが繰り返す確率

15種類は再発しなかった (全32種中)

エラータイプ	再発率
OSError	71.43%
django.db.utils.IntegrityError	57.69%
IndentationError	50.00%
sqlite3.IntegrityError	50.00%

頻度は比較的少なかった (RQ2参照) が再発率は1位

調査結果 | RQ3 (解決しづらいエラーは?)

再発率 (RR)を計算

- 同じタスク内で同じエラーが繰り返す確率

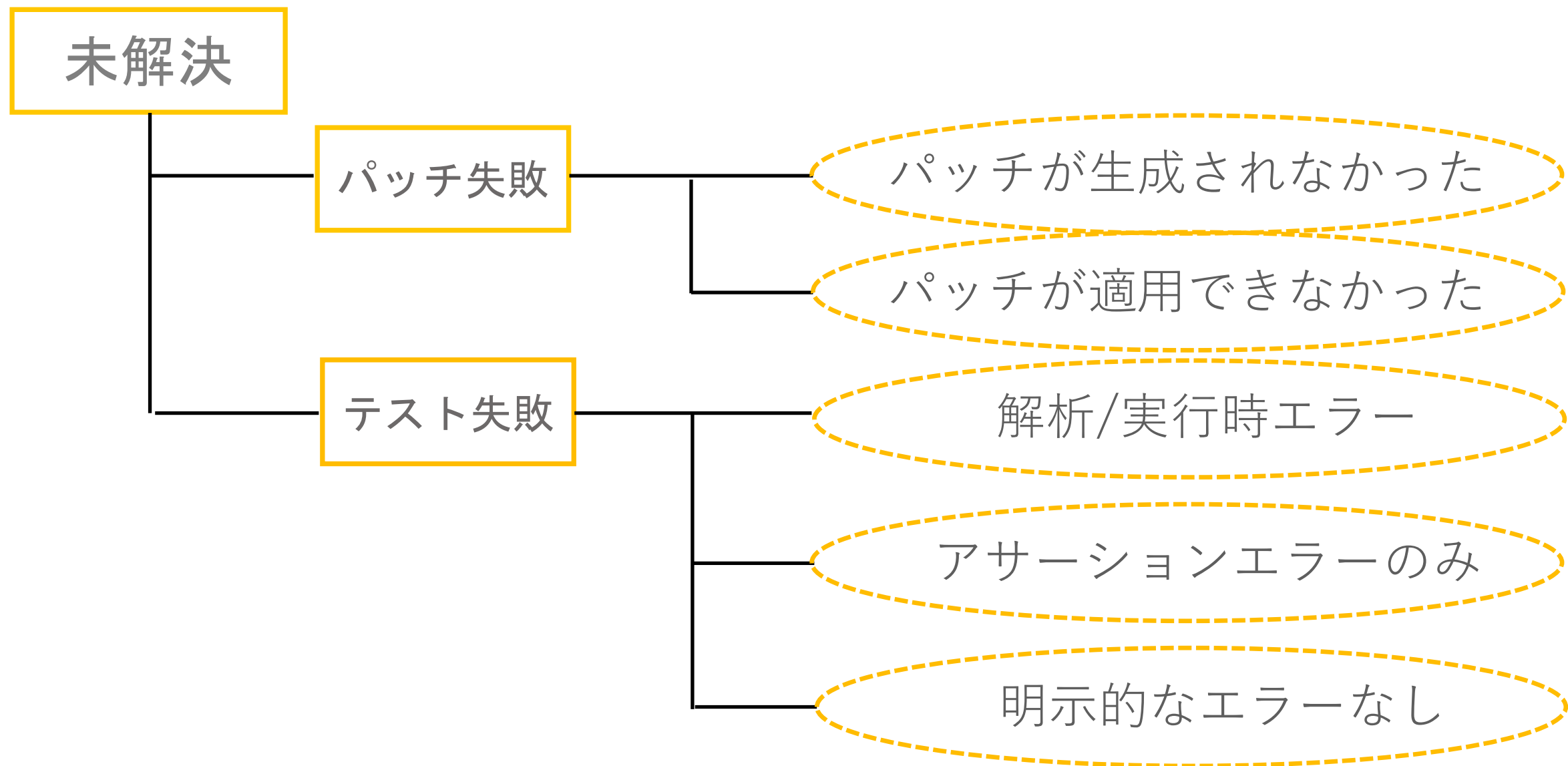
15種類は再発しなかった (全32種中)

エラータイプ	再発率
OSError	71.43%
django.db.utils.IntegrityError	57.69%
IndentationError	50.00%
sqlite3.IntegrityError	50.00%

DB系のエラーは頻度も高く、再発率も高い

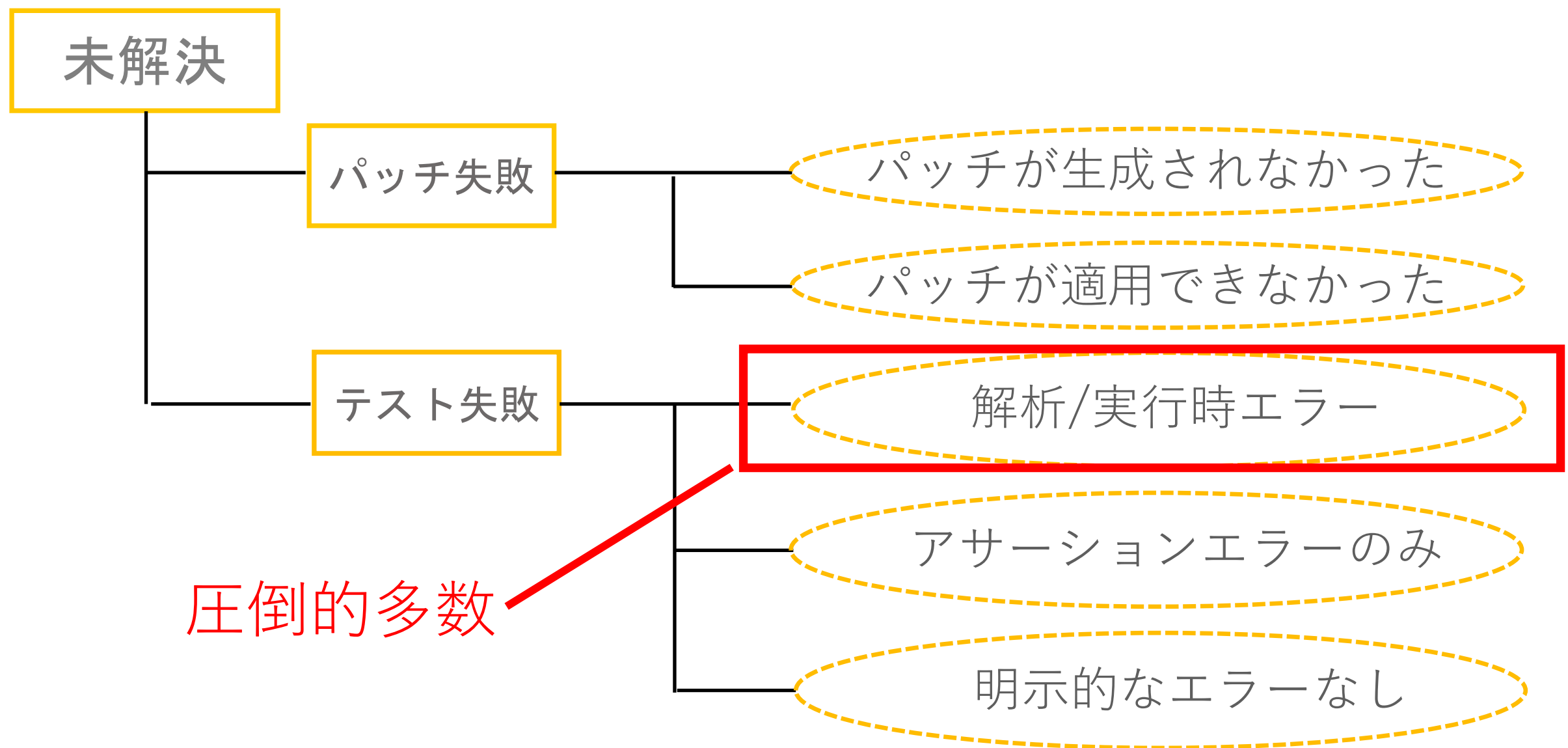
調査結果 | RQ4 (最終結果のエラー)

未解決タスクの分類



調査結果 | RQ4 (最終結果のエラー)

未解決タスクの分類



調査結果 | RQ4 (最終結果のエラー)

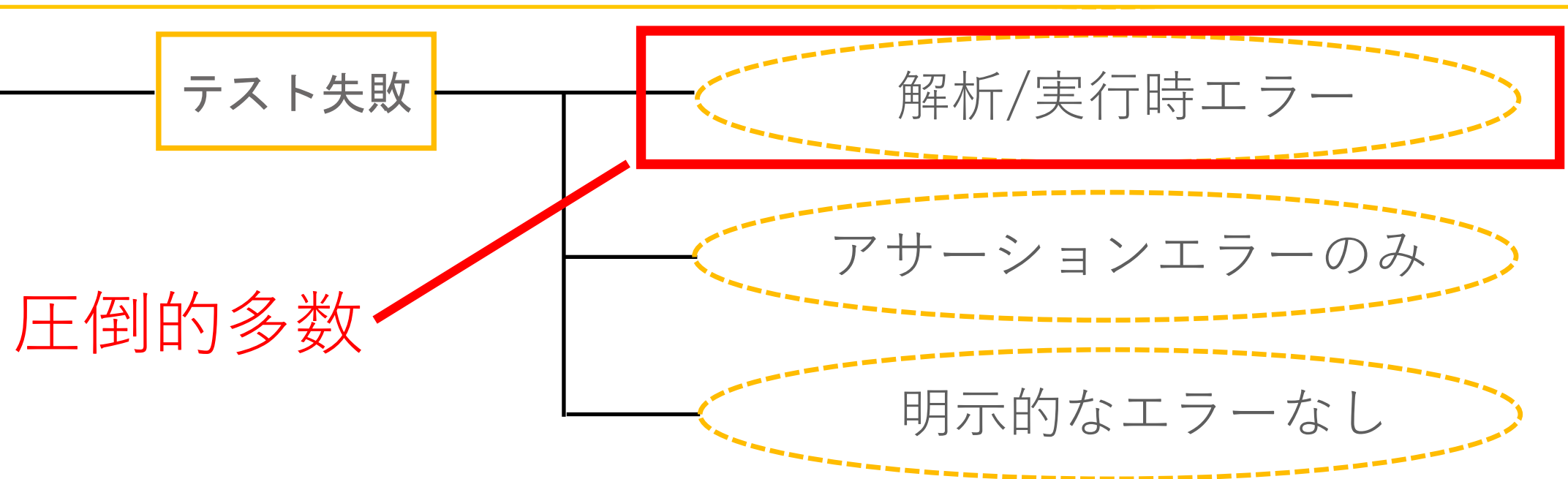
未解決タスクの分類

W&B Programmerでは 未解決177件中 147件

CodeShell では 未解決279件中 237件

その他エージェントでも同じ傾向

コードが動かない状態で提出しているケースが多い



調査結果 | RQ4 (最終結果のエラー)

W&B Programmerでは 未解決177件中 147件

CodeShell では 未解決279件中 237件

その他エージェントでも同じ傾向

コードが動かない状態で提出しているケースが多い

解決フェーズと同じエラーが出たもの (件数上位)

- TypeError 10回
- ModuleNotFoundError 9回
- AttributeError 6回
- NameError 6回

まとめ

エージェントの「解決プロセス」を分析

エラー頻度の増加は**解決率の低下**と**コストの増加**を招いた

環境構成・内部検証のエラーが頻発した

頻度は高くないが再発率が高い（修正しにくい）エラーを特定