Why Do Developers Engage with ChatGPT in Issue-Tracker? Investigating Usage and Reliance on ChatGPT-Generated Code

Joy Krishan Das, Saikat Mondal, Chanchal K. Roy, 10 December 2024.

SANER 2025 Research Papers

背景

論文の日本語訳

開発者が課題追跡ツールでChatGPTを利用する理由とは？
ChatGPT生成コードの使用状況と依存度を調査

概要

ChatGPTなどのLLMは様々な問題解決に利用されているが，その全容の実証的研究は不足している.

➤ChatGPTでのコード生成に開発者達はどれほど依存しているのかという所に重点を置いて実証的研究を行った.

課題

本論のリサーチクエスチョン

RQ1 :課題管理システム（ITS）で開発者が使用する主な目的とは？

開発者はITSでChatGPTを最も活用しているため，ニーズが理解できるはず.

RQ2 :開発者がChatGPTとやりとりする主な分野は？

RQ1を補完するためのRQであり，ニーズに対応するツールの方向性を示す.

RQ3 :ChatGPT生成コードへの依存度はどの程度か？

依存度を理解することで，実際のコードベースに統合しているかどうかが分かる.

RQ4 :用途や分野ごとの，ChatGPTの解決案に対する満足度は？

感情分析を用いて会話内の感情を指数移動平均（EMA）で測定する.

研究手法

データ収集

GitHub Issues内で共有されたChatGPTとの会話リンクを収集した.

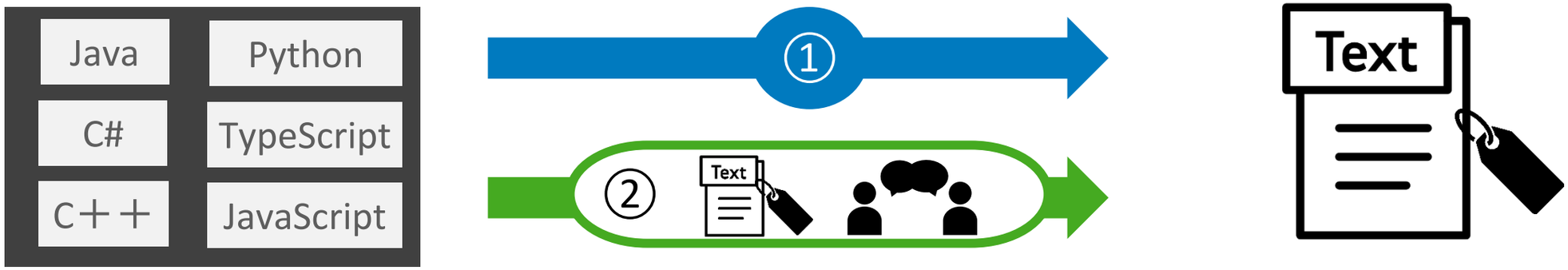
- リンク切れ，翻訳エラーが出たものは例外.
- 使用言語は6つに指定（Java，C++，C#，Python，TypeScript，JavaScript）

➤1012件の問題と1152件の会話リンクが収集された.

研究手法

ChatGPTの使用法の分類（RQ1）

Cruzesらのテーマ抽出法で会話のラベル分けを行った. [1]



①６つのグループから
ランダムに標本を抽出

②別の会話をラベル付け
して共通認識を構築

③抽出した会話の中から
40件をラベル付け

➤しかし，ラベリングの精度は67%にしか達しなかった.

精度を上げるためにラベルを統合し，再度分類を行った.

・意味的に同義と見なせるラベルを統合.

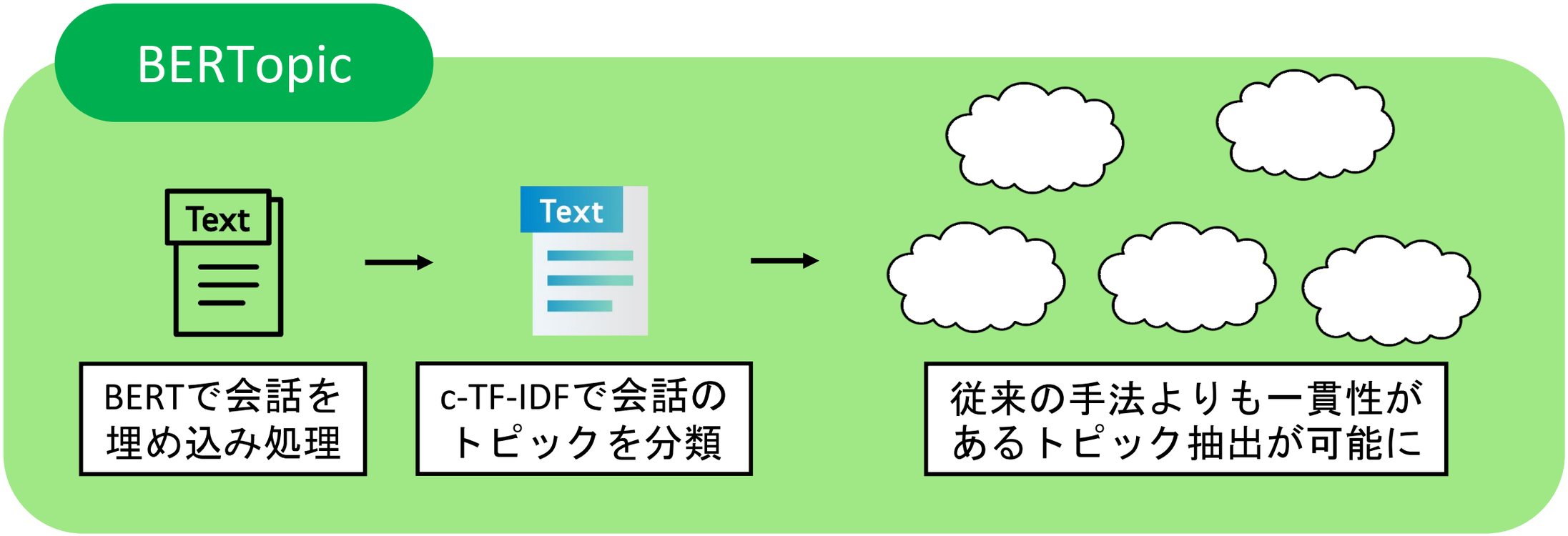
➤ラベリングの精度は92.5%に達した.

[1] D. S. Cruzes and T. Dyba, “Recommended steps for thematic synthesis in software engineering,” in 2011 International Symposium on empirical software engineering and Measurement. IEEE, 2011, pp. 275–284.

研究手法

ChatGPTの主な議論分野の特定（RQ2）

BERTopicを使用し，ChatGPTとの会話から主要トピックを抽出.



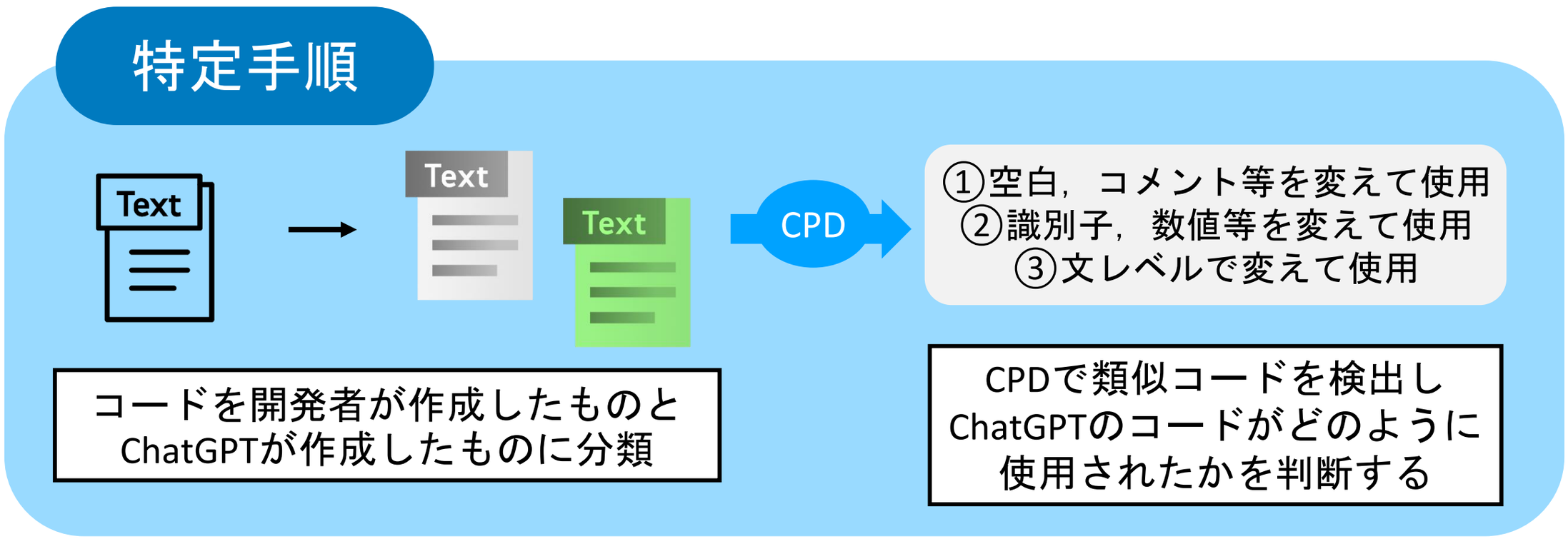
➤まとめられたトピックを，その後手動でラベル付けを行った.

➤9つの議論領域にまとめることができた.

研究手法

ChatGPT生成コードの使用状況の特定（RQ3）

CPDというコードクローン検出ツールを用いて実際に生成されたコードがどのように使用されたかを調査した.



研究手法

ChatGPTの応答に対する満足度評価（RQ4）

感情分析モデルを使用し，開発者の会話中の感情指標を分類した.

・EMA（指数移動平均）によって感情の傾向を測定した.

分析手法

$$EMA_t = \alpha S_t + (1 - \alpha) EMA_{t-1}$$

$$\alpha = \frac{2}{n+1}$$

S_t :時点tにおける感情スコア
α :会話の重要性を表す平滑化関数

感情スコアS_tと平滑化係数αから
時点tでの EMAを求める

EMAが負の場合：悪い感情を持つ
EMAが正の場合：良い感情を持つ
絶対値が大きいと，感情の振れ幅が大きい.

研究結果

RQ1 :課題管理システムでのChatGPTの主な利用用途

以下の５つに分類され，アイデア出しのタスクが最も多く，検証タスクが最も少なかった.

- アイデア出し
- コード生成
- コード理解
- デバッグ
- 検証

研究結果

RQ2 :開発者がChatGPTとやりとりする主な分野は？

以下の９つに分類され，バックエンド開発が最も多く議論され，テストが最も少なかった.

- バックエンド開発
- データ分析
- コードリファクタリング
- テスト
- デバッグ
- 自動化
- フロントエンド開発
- ドキュメーション
- 依存関係の管理

研究結果

RQ3 :ChatGPT生成コードへの依存度はどの程度か？

分析の結果，ChatGPT生成コードは**全体の約12%**の課題解決に利用されていることが判明した.

また約7割は既存コードのリファクタリングなどに使用されていて，新規コード作成などには使用されていなかった.

研究結果

RQ4 :用途ごとのChatGPTの案に対する満足度は？

全体の感情

約22%がポジティブな感情を，約78%はネガティブな感情を示した.

使用用途別の感情

示されたポジティブな感情はこのようになった.

- コードリファクタリング：47.3%
- データ分析：35.9%
- デバッグ：7.84%
- 自動化：12.7%
- バックエンド開発：16.1%

➤ChatGPTの満足度は分析ツールなどで高いということが示された.

考察

最新であるほど信頼性が高くはない

GPT-4とGPT-3.5を比べると，GPT-3.5の生成コードの方が活用されていることが分かった.

満足度も高いわけではない

満足度の観点でも，GPT-3.5の方がポジティブな感情の割合が高いという結果になった.

不満点は様々

コードが冗長，情報過多，具体性がない，複雑な問題に弱いことと，色々な不満点が考えられる.

結論

ブレインストーミング，コード生成に最も利用される

内容は**バックエンド開発**に関するものが多く見られた.

データ分析やリファクタリングで満足度が高い

対してデバッグや自動化タスクの回答には不満点だらけであった.

➤開発者がChatGPTで満たされていないニーズの指標となる.

所感

満足度の目安が少し曖昧

GPT-3.5の方が満足度が高い = 無料であることが加味されてなさそう.

ChatGPTとGUIテストは合わない？

RQ1，RQ2で検証，テスト関連にChatGPTが最も使用されていないことが分かることから考えられる.

➤逆にまだ使われてないだけなのか.