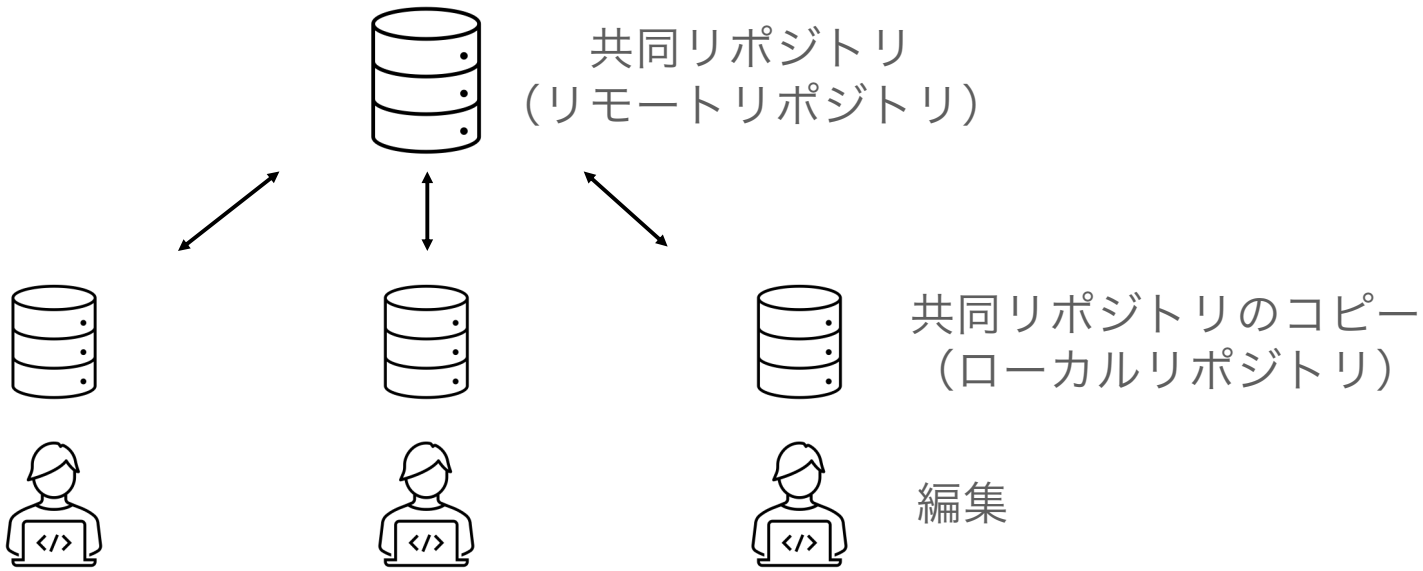
対照学習を用いた変更理由を含むコミットメッセージの生成

オープンソースソフトウェア工学研究室
S2420142 山村慎太郎

Git

- 共同でのソフトウェア開発を効率的に行うための，分散型バージョン管理システム



- チーム開発におけるソースコード管理のため，ソフトウェア開発の現場で広く利用されている

コミット

Gitにおいて，ファイル・ソースコードの変更を，ローカルリポジトリに記録すること

手順

1. git add ファイル名 でファイルの変更をステージングエリアに追加
2. git commit –m “変更内容の説明” で変更をコミット

コミットで変更を記録すると，

・変更履歴の確認

・過去のコミットの状態に戻す(ロールバック)

などが可能となる

コミットの際，通常，”変更内容に関する説明”を付与する

コミットメッセージ

- コミットを行う際に付与する変更内容に関する説明

例

コード差分	コミットメッセージ
+ configureFeatures(builder, this.jacksonProperties.getEnumDatatype()); + configureFeatures(builder, this.jacksonProperties.getJsonNodeDatatype());	Add properties for configuring EnumFeature and JsonNodeFeature

- コミットメッセージを書くことで,
  - 変更履歴の可読性の向上
  - ロールバックの容易性

をもたらし，チーム開発の効率化に貢献

コミットメッセージの現状

コミットメッセージを記述するのには，時間や労力がかかる

原因：コミット内容が複雑，簡潔に要約するのが難しい　など

その結果，

・メッセージ内容が完全に空白

・メッセージ内容が数語しか書かれてない（例）

コード差分	コミットメッセージ
- import_paths = [] - - for root, dirs, files in os.walk("./imports"): - for file in files: - if file.endswith(".py"): - import_paths.append(os.path.join(root, file))	cleanup

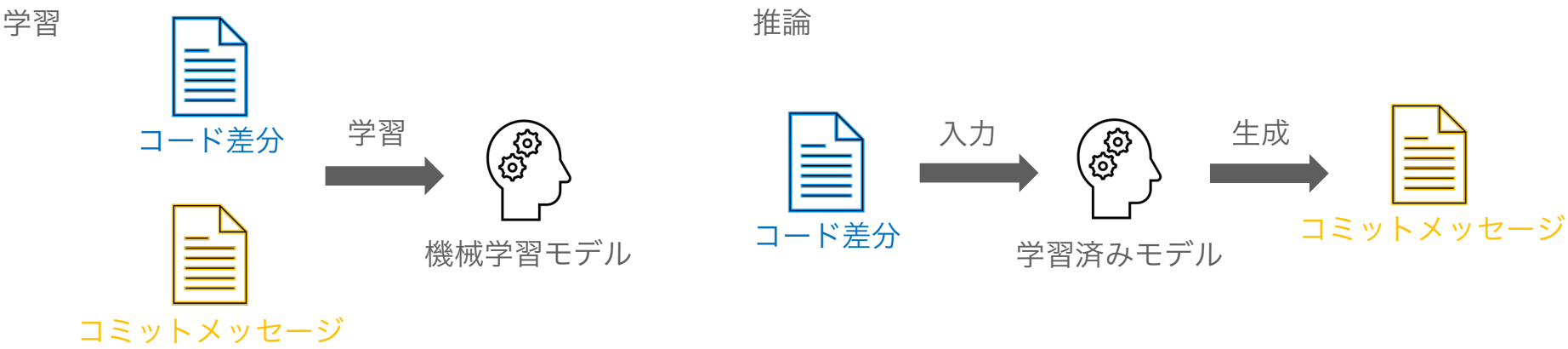
のような，品質の低いメッセージが書かれる場合がある

そこで，コミットメッセージの自動生成が研究される

コミットメッセージの自動生成

- コミットメッセージを自動で生成することで，メッセージを書く手間が省ける

例：学習ベース



- 課題：内容が不十分なメッセージがほとんど

要因：データセットについて，

・OSS projectの44%のメッセージは変更内容や変更理由を含まない

・メッセージの内容を考慮・区別して学習していない

自動生成の課題

変更理由を含むメッセージが生成されない

含まない例：メッセージが変更内容のみ

コード差分	コミットメッセージ
- const response = await fetch('/api/profile', { + const response = await fetch('/api/users/profile', {	Update user profile UI

変更理由が重要な理由

・共同開発者が変更内容を理解するため

・変更を行なった開発者がプロジェクトを去った後,
その変更の意図を知る唯一の情報源となりうるため

そこで，変更理由を生成できるような学習を行う

本研究の目的・アプローチ

目的

変更理由を含むコミットメッセージの自動生成

アプローチ

・データを以下のように区別して学習

〇変更理由を含むコミットメッセージ

〇変更理由を含まないコミットメッセージ

・対照学習という手法を用いて，より変更理由を含むコミットメッセージの特徴を学習

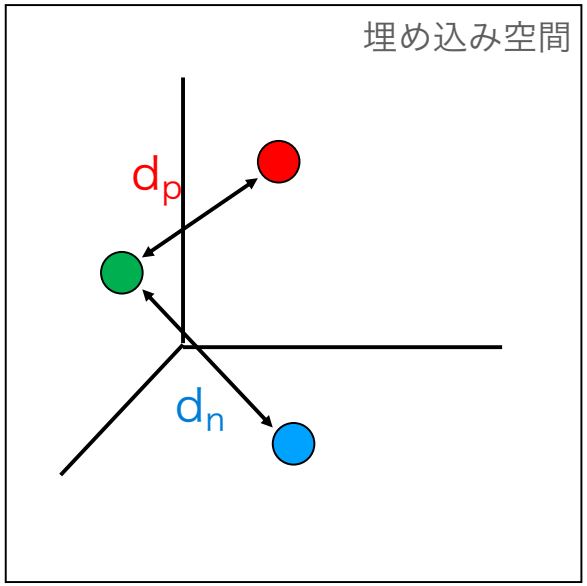
対照学習の動機

機械学習・表現学習手法の一つ

・同じクラスに属するデータ同士を近くする

・異なるクラスに属するデータを遠ざける

ように，データの表現を学習させる



$$L = [d_p - d_n + \alpha]_+$$

（Lが最小となるよう最適化）

コード差分に対し，

・変更理由を含むCMを近く

・変更理由を含まないCMを遠く

なるように表現を学習することで，変更理由を含むCMをより生成しやすくなるのでは？

進捗と予定

現状

そもそも実現可能か，仕組みはどうなっているのか

がわかっていない状態

→勉強が必要

進捗

LLMや対照学習について勉強中

でも必要な理解レベルにはまだまだ到達していない

予定

5月中に本手法の仕組みを理解・図示・説明できるように

6月中に実験設定やデータセットの調査