

技術的負債に関する課題票 のナイーブベイズを用いた テキスト分類(仮)

発表：山村慎太郎

技術的負債(Technical Debt : TD)

ソフトウェア開発において、時間的制約などにより設計や実装の質を妥協する事象

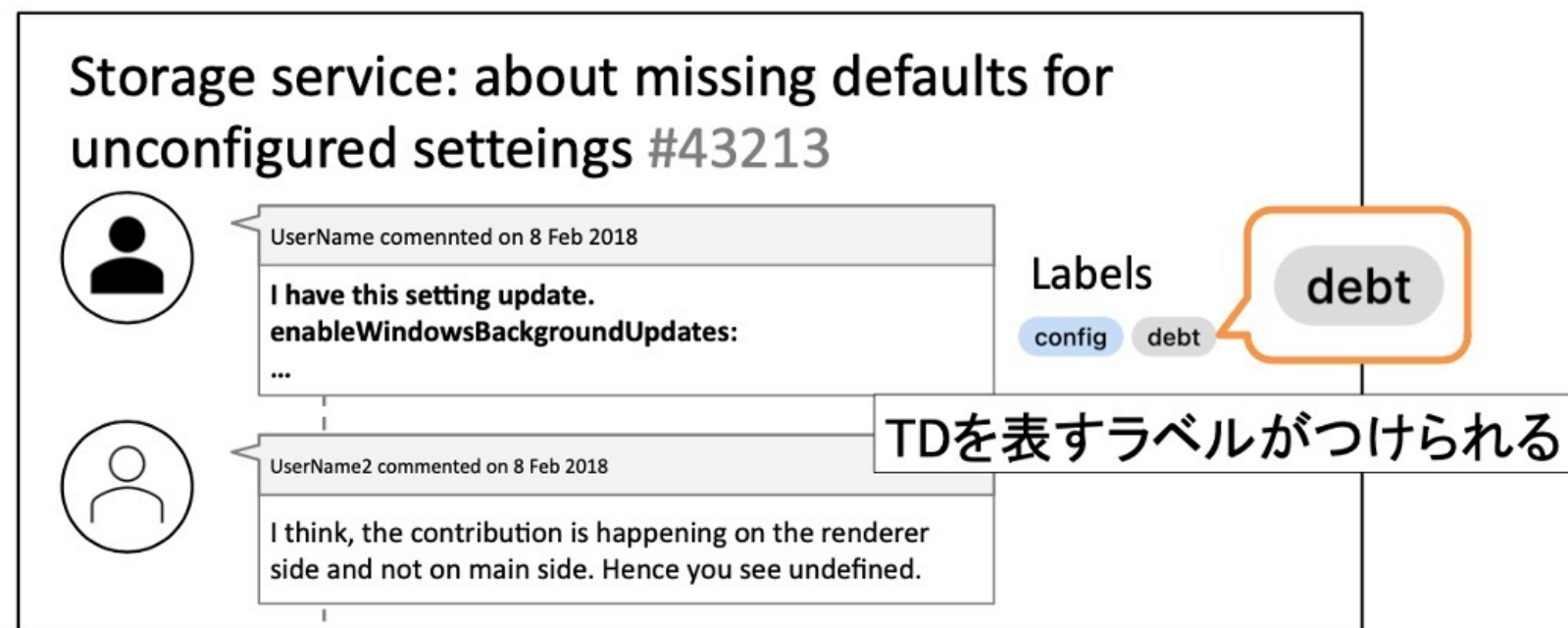
- ・スパゲッティコード
- ・コーディング規約に反するソースコード など

TDが蓄積すると除去のための保守コストが増加

⇒早期除去には、TDの管理が重要

課題票として管理されるTD

TDに関する課題票を，課題管理システムで管理する場合がある

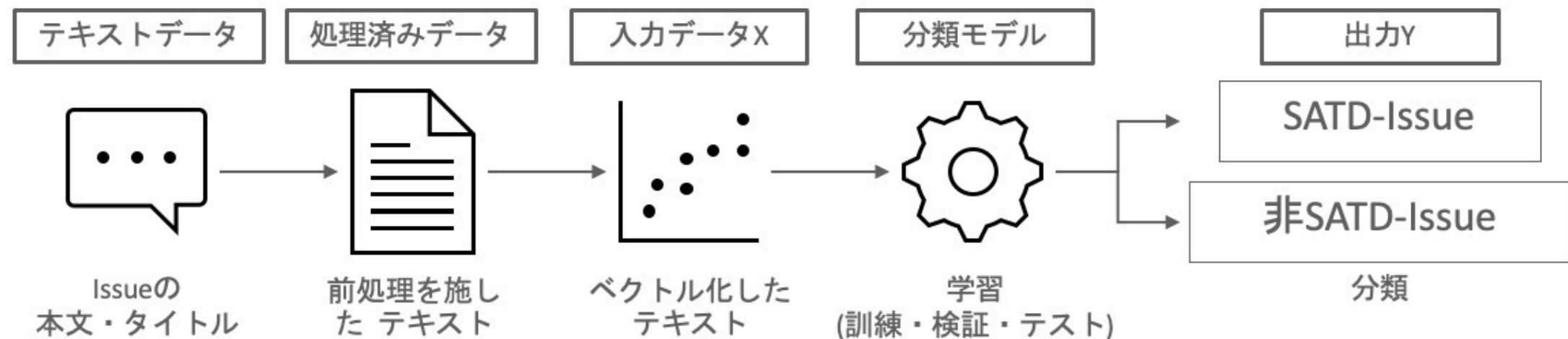


課題票を登録する段階では，TD関連であることが明示されない場合がある

—TDと通常の不具合とでは対応が異なるため，登録時点でTDであるか区別できているのが望ましい

先行研究

課題管理システムに登録された時点で、課題票の内容がTDに関するものかどうかを判別する分類モデルを構築



ベクトル化にはBoW, TF*IDF, Word2Vec, FastText, Glove

分類にはランダムフォレスト, ロジスティック回帰, サポートベクターマシンを使用

先行研究における課題点

レンマ化の際、辞書外の単語は全て「oov」に置換していた点

→置換せずに分類を行った際の分類精度は？

分類モデルとしてランダムフォレスト、ロジスティック回帰、SVMのみを用いていた点

→他の代表的な分類アルゴリズムであるナイーブベイズを用いた場合の分類精度は？

ベクトル化手法FastTextにおいて、未知語への対応ができていなかった

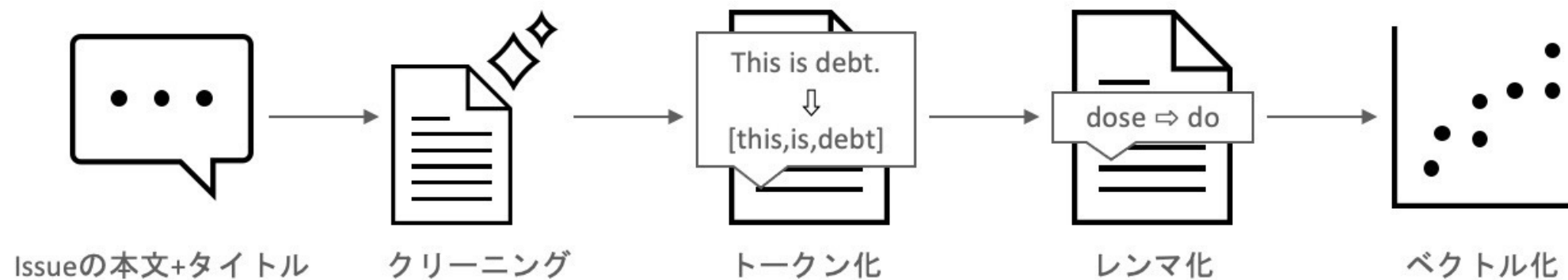
→未知語に対応したFastTextを用いた場合の分類精度は？

目的

- | TDに関する課題票の分類において、先行研究の課題点を改善した結果、分類精度が向上するかを分析
- | 課題の改善がそれぞれ分類精度にどのような影響を及ぼすかを分析

評価実験①

テキストデータの前処理



辞書外の単語を置換する場合
としない場合に分ける

利用する単語のベクトル化手法

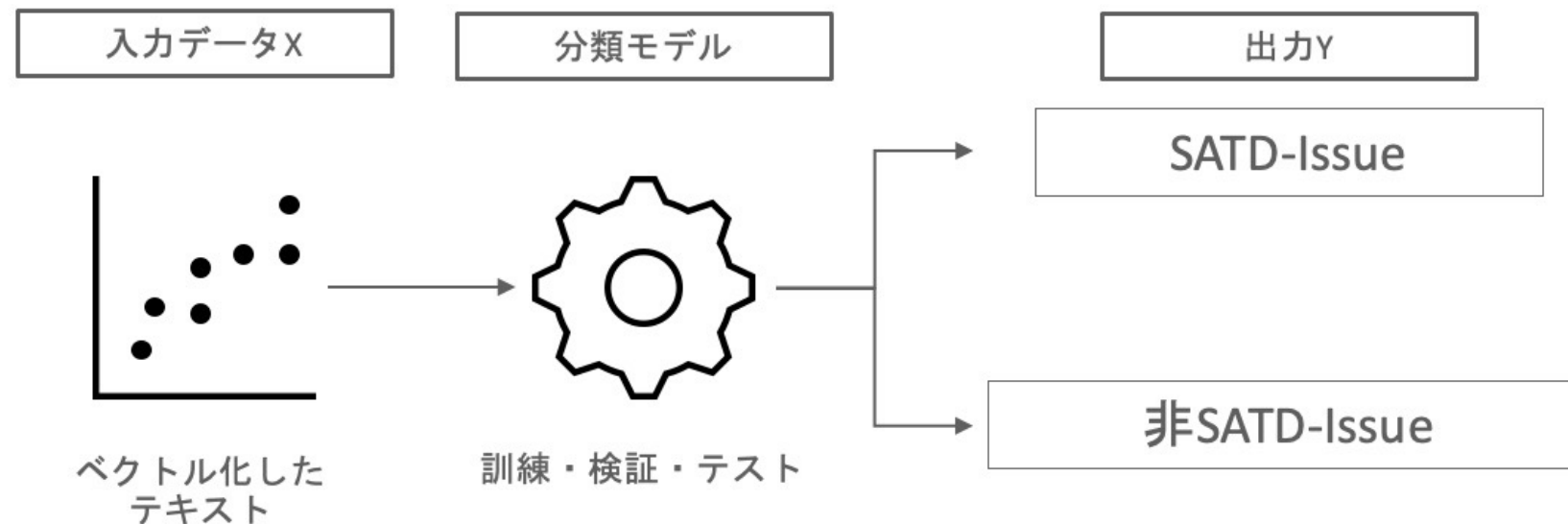
- BoW, TF*IDF, Word2Vec, FastText, Glove

Stack OverflowまたはWikipediaから
取得したコーパスで学習したモデル

Wikipediaから
取得したコーパスで学習したモデル

- FastTextは未知語に対応

評価実験②



機械学習アルゴリズム

- ・ ランダムフォレスト
- ・ ロジスティック回帰
- ・ サポートベクターマシン
- ・ ナイーブベイズ

OOV置換あり・なし：2種類
×
ベクトル化手法：6種類
×
機械学習アルゴリズム：4種類
||
分類モデル：48種類
どの分類モデルの性能が良いかを調査

ナイーブベイズ

ベイズの定理を元にした教師あり学習アルゴリズム 特徴

- ・ 単語の出現確率に基づいて予測を行う
- ・ アルゴリズムがシンプルで、処理が高速
- ・ 単語はそれぞれ独立であると仮定する

$$P(X|Y) = \frac{P(X) \times P(Y|X)}{P(Y)}$$

ベイズの定理

TDに関するIssue特有の単語が頻出するのであれば、
高い精度が期待できる？

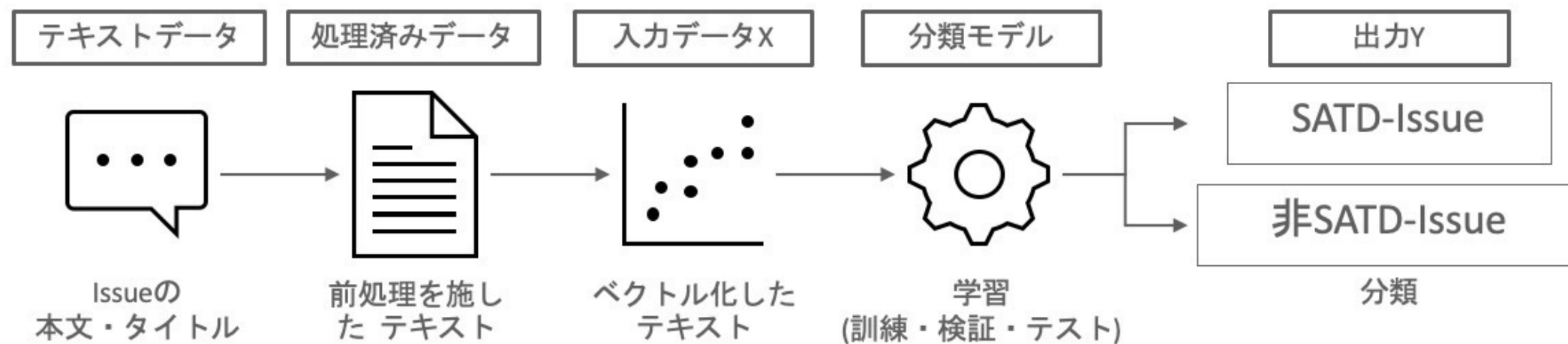
データセット

- GitHub上のスター数上位5000件のうち,
- ・ TDに関するIssueをよく扱う4つのプロジェクトの内,
- ・ 2回以上報告経験のある開発者の解決済みIssue

オーナー / リポジトリ名	SATD の label	非 SATD-Issue 数	SATD-Issue 数
influxdata / influxdb	kind/tech-debt	8,514	120
saleor / saleor	technical debt	2,230	108
nextcloud / server	technical debts	7,255	83
microsoft / vscode	debt	79,829	2,044
合計	-	97,828	2,355

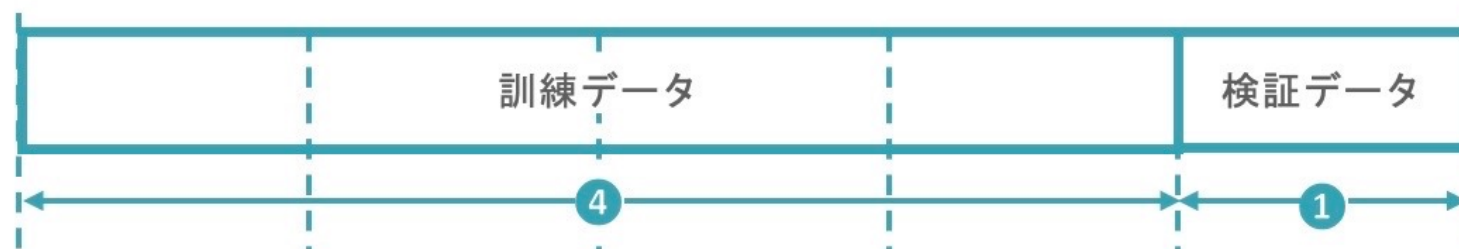
Undersampling, SMOTEにより学習データ数を調節
—データ不均衡が原因でナイーブベイズの精度がまともに出ない

分類モデルの作成



データの正規化：
$$X' = \frac{X - x_{min}}{x_{max} - x_{min}}$$

層化5分割交差検証



- ・ 検証データ交替5回×再分割して20回=合計100回学習

取り組むこと・進捗

やること

田口先輩の実験コード(卒研用)が残っているので、それを元にoov処理, NB, FastTextの実装を行う

進捗

実装は概ね完了(oov有無ver, NB, FastText)

細かい点を確認・改善して実行すれば実験終了

今後の予定(10月～?)

- ・ 実験結果を出す
- ・ 卒論に取り掛かりつつ, 結果の分析・考察や, 知識の補完