

# A Deep Multitask Learning Approach for Requirements Discovery and Annotation from Open Forum

## マルチタスク深層学習を用いた オープンフォーラムからの要求発見と注釈

Mingyang Li<sup>1</sup>, Lin Shi<sup>1</sup>, Ye Yang<sup>2</sup>, Qing Wang<sup>1</sup>

<sup>1</sup>University of Chinese Academy of Sciences, Beijing, China

<sup>2</sup>Stevens Institute of Technology, Hoboken, NJ, USA

35th IEEE/ACM International Conference on Automated Software Engineering (ASE 2020), pp. 336-348.

ソフトウェアを継続的に改善していくためには、  
ユーザから得られるフィードバックや新機能の提案を取り入れる必要

しかし、チャット、電子メール、フォーラムなどに投稿される  
大量のテキストデータから、手動で要求を抽出するのは、非常に労力がかかる

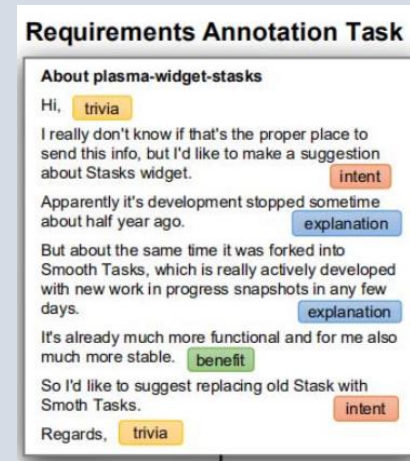
そのため、テキストマイニング、機械学習などの技術を使った、  
自動で要求を発見、分析するアプローチが提案されている<sup>[18][51][62]</sup>

[18] Qiao Huang, Xin Xia, David Lo, and Gail C. Murphy. 2018. Automating Intention Mining. IEEE Transactions on Software Engineering PP, 99 (2018), 1–1.

[51] Lin Shi, Celia Chen, Qing Wang, Shoubin Li, and Barry W Boehm. 2017. Understanding feature requests by leveraging fuzzy method and linguistic analysis. automated software engineering (2017), 440–450.

[62] Radu Vlas and William N. Robinson. 2011. A Rule-Based Natural Language Technique for Requirements Discovery and Classification. In in Open-Source Software Development Projects. Proceedings of the 44th Hawaii International Conference on Systems Science.

# 背景 | OSSにおける要求抽出プロセス



①メンテナがIssueを閲覧，  
要求が書かれたIssueに  
“feature request”などのラベルを付与  
→ 要求発見

②要求が書かれたIssueの  
提案をレビューし，  
文レベルで意味的な注釈  
→ 要求注釈

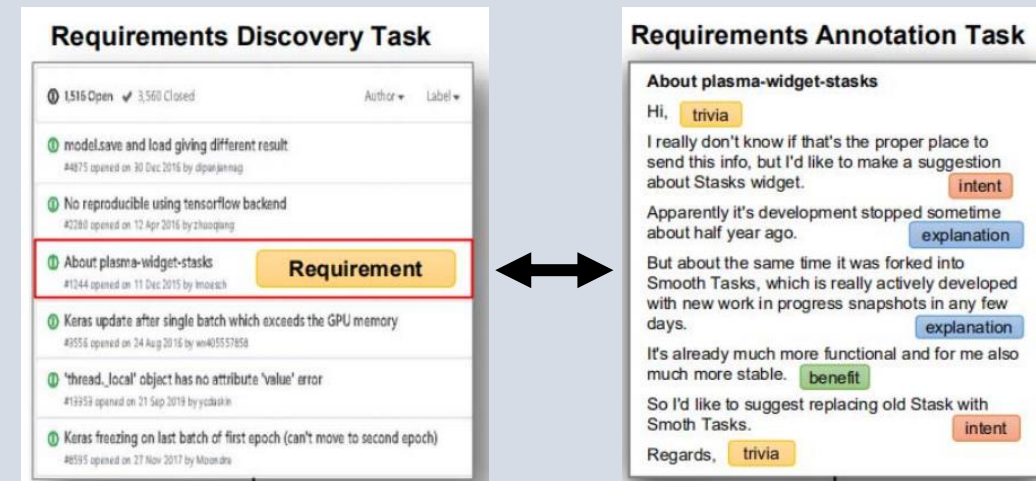
これらのプロセスを自動化することにより，ユーザが投稿した構造化されていない  
大量のテキストから，要求に関する情報を効率的に抽出できる<sup>[51]</sup>

[51] Lin Shi, Celia Chen, Qing Wang, Shoubin Li, and Barry W Boehm. 2017. Understanding feature requests by leveraging fuzzy method and linguistic analysis. automated software engineering (2017), 440–450.

# 背景 | 要求発見と要求注釈の相関

既存研究では<sup>[18][51]</sup>，要求発見/注釈タスクを個別に解決  
しかし，これらのタスクには**相関する情報**が存在

- 要求発見が，要求注釈に役立つ例  
要求が書かれたIssueであれば，  
意図(Intent)注釈が含まれる可能性
- 要求注釈が，要求発見に役立つ例  
注釈の順番が，意図(Intent) → 説明 (Explanation)  
→ 利点(Benefit) であれば，要求が書かれたIssue  
である可能性



**マルチタスク学習**を利用し2つのタスクを共同で学習，自動化の性能向上を目指す

[18] Qiao Huang, Xin Xia, David Lo, and Gail C. Murphy. 2018. Automating Intention Mining. IEEE Transactions on Software Engineering PP, 99 (2018), 1–1.

[51] Lin Shi, Celia Chen, Qing Wang, Shoubin Li, and Barry W Boehm. 2017. Understanding feature requests by leveraging fuzzy method and linguistic analysis. automated software engineering (2017), 440–450.

# 目的とアプローチ

## 目的

OSSプロジェクト上の大量のIssueからの**要求発見/注釈の自動化**を行う

**マルチタスク深層学習**モデルDEMAR<sub>(DEep Multitask LeArning for Requirements Discovery and Annotation)</sub>を構築

## アプローチ

(1) Issueのテキストから要求発見/注釈を行うための**データセット**を作成

(2) マルチタスク深層学習を行うための**モデル構築**

# データセット作成

8つのApacheプロジェクトからIssueを収集

- 要求発見タスク  
”feature request”, ”enhancement”ラベルが付いたIssue → 要求Issue  
それ以外のIssueから要求Issueと同数をランダムサンプリング → 非要求Issue  
要求Issue：573件，非要求Issue：494件 (本文が空白のIssueを除外)

- 要求注釈タスク  
先行研究<sup>[51]</sup>で定義された7つのカテゴリで文に手動でラベル付け

| カテゴリ                  | 定義              |
|-----------------------|-----------------|
| 意図 (Intent)           | 機能を改善するアイデアやニーズ |
| 利点 (Benefit)          | 提案された機能のメリット    |
| 欠点 (Drawback)         | 現在の機能の欠点        |
| 例 (Example)           | 提案された機能の動作例     |
| 説明 (Explanation)      | 提案された機能に関する詳細説明 |
| 現状 (Current Behavior) | 機能の現在の動作        |
| その他 (Trivia)          | 要求に無関係          |

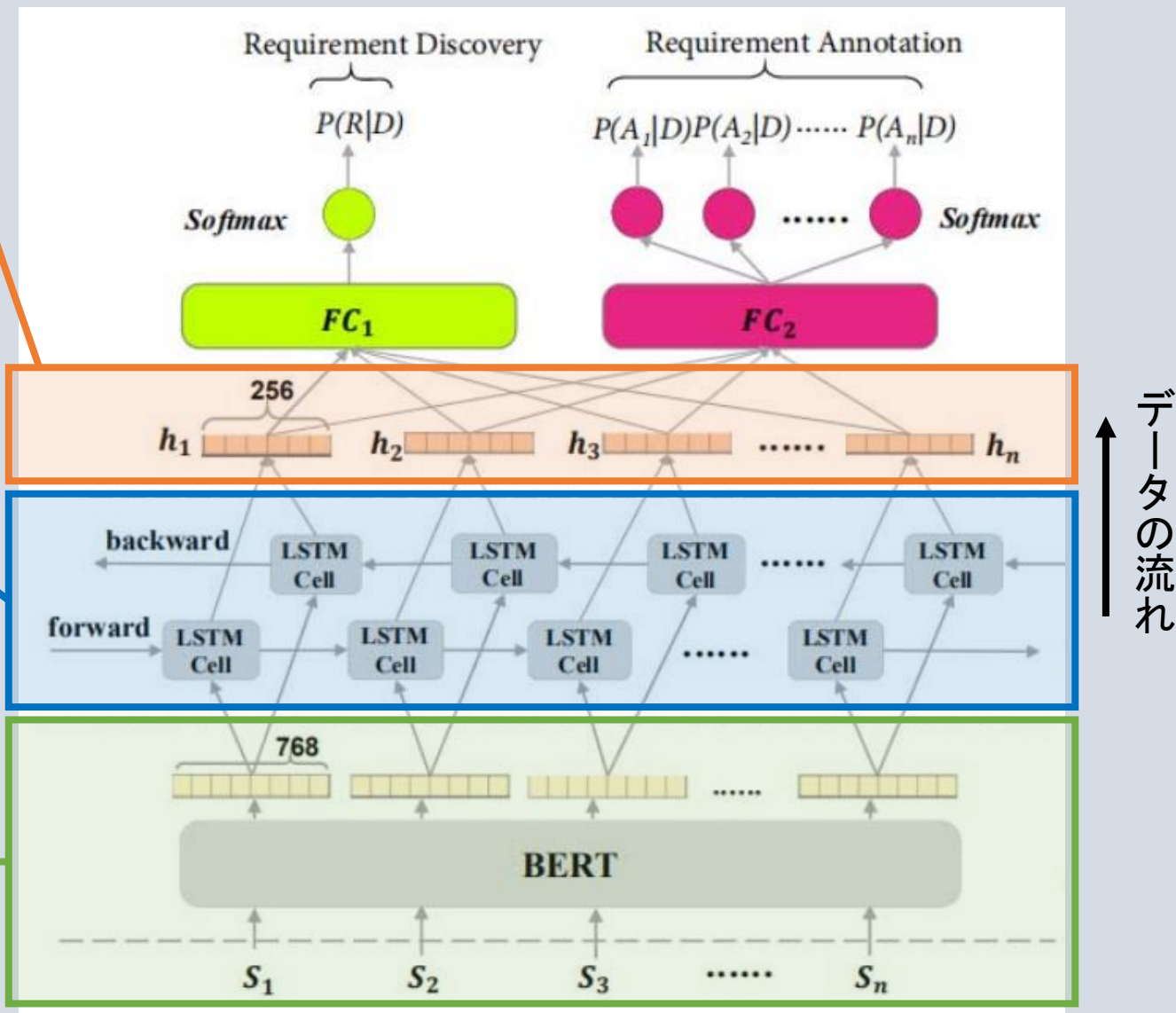
[51] Lin Shi, Celia Chen, Qing Wang, Shoubin Li, and Barry W Boehm. 2017. Understanding feature requests by leveraging fuzzy method and linguistic analysis. automated software engineering (2017), 440–450.

# モデル構築 | タスク共通部分

入力した文 $s_1, s_2, \dots, s_n$ に対応する  
LSTMの出力 $h_1, h_2, \dots, h_n$ を結合し、  
各タスク共通で利用する特徴表現とする

双方向LSTMを利用して、  
コンテキスト情報を256次元の  
ベクトルに埋め込む

BERTを利用して、1文ごとに  
768次元のベクトルに埋め込む



データの流れ

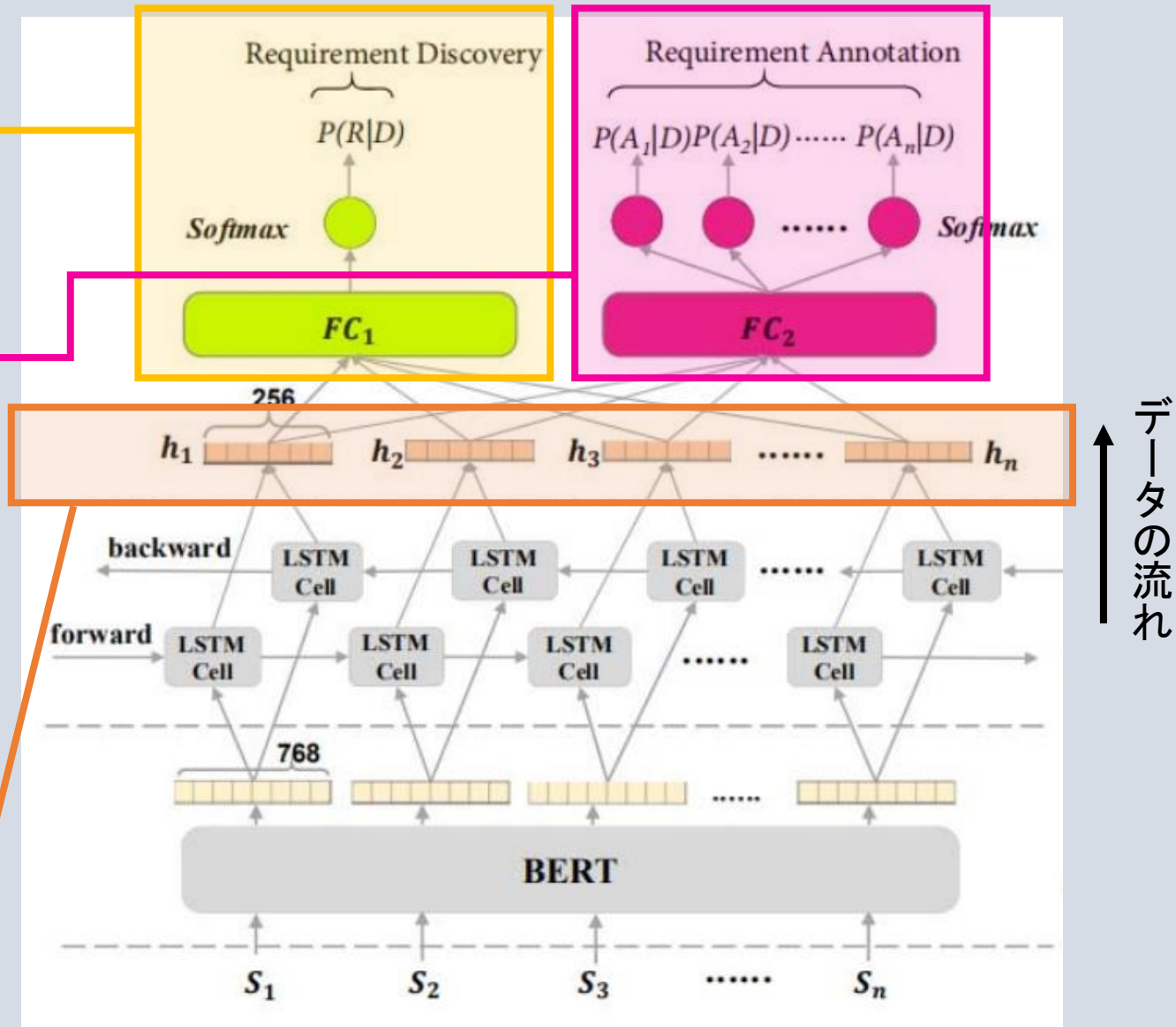
# モデル構築 | タスク個別部分

共通の特徴表現を入力とし、  
Issueを要求/非要求に2値分類

共通の特徴表現を入力とし、  
各文を7種類の注釈にマルチラベル分類

それぞれのタスクでの損失の平均値を  
ネットワークの損失として最適化

共通の特徴表現には、要求発見/注釈  
の両方のタスクを解くための  
情報が埋め込まれる



構築したモデルの性能を評価するため、RQを設定

RQ1：シングルタスク学習と比べてマルチタスク学習は性能が高いか？

要求発見/注釈の自動化におけるマルチタスク学習の有効性評価

RQ2：既存手法と比べて提案手法の性能が高いか？

既存の要求発見/注釈の自動化手法，および広く利用されている機械学習分類モデルと比べたときの，提案手法の有効性評価

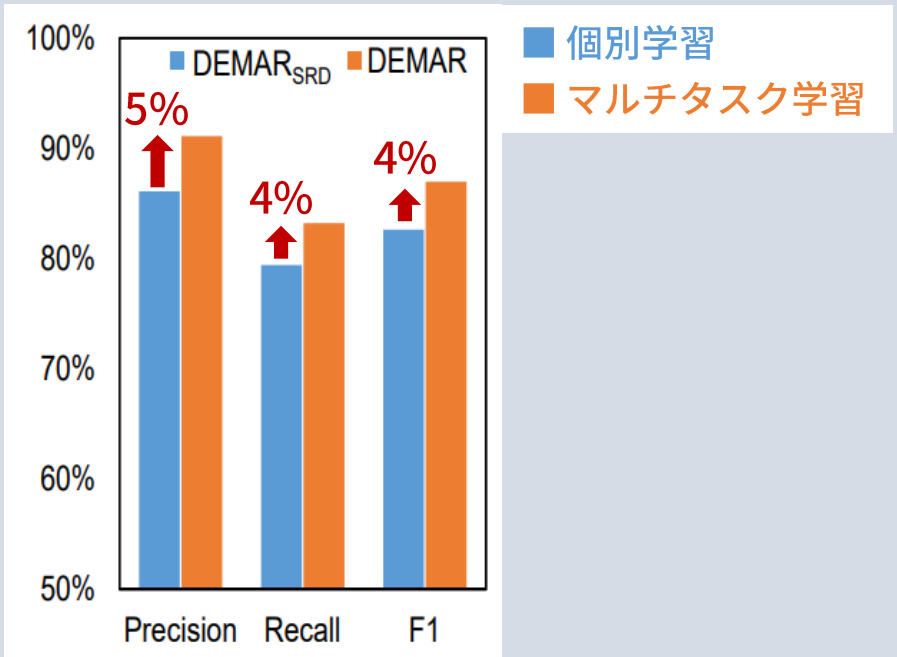
# RQ1：シングルタスク学習と比べてマルチタスク学習は性能が高いか？

## 方法

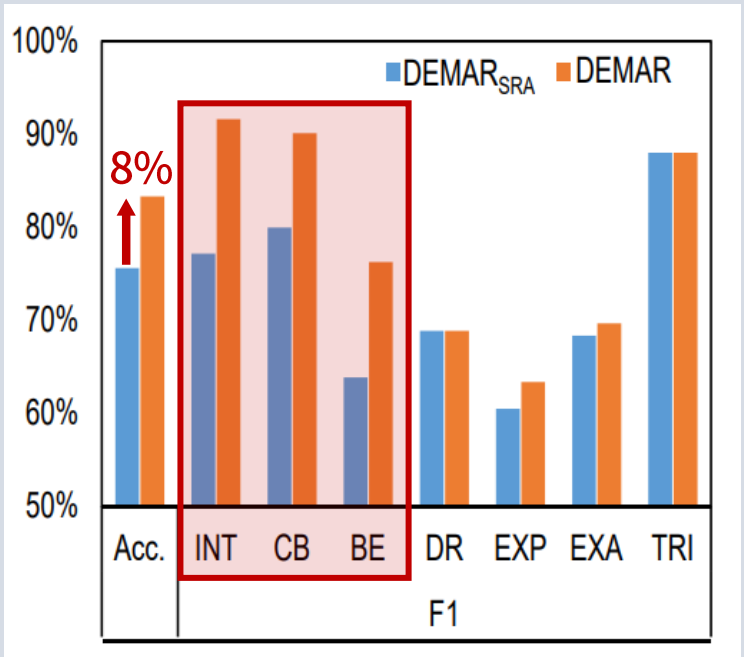
要求発見/注釈を個別で学習するモデルを作成し，マルチタスク学習モデルと比較

## 結果

### ・ 要求発見



### ・ 要求注釈



カテゴリ別では，

- ・ 意図 (INT)
  - ・ 現状 (CB)
  - ・ 利点 (BE)
- F1が向上

理由：Issueが要求/非要求かによって出現確率が変わる

Answer：特定の注釈カテゴリがタスク間で相関，マルチタスク学習で両タスクの分類性能向上

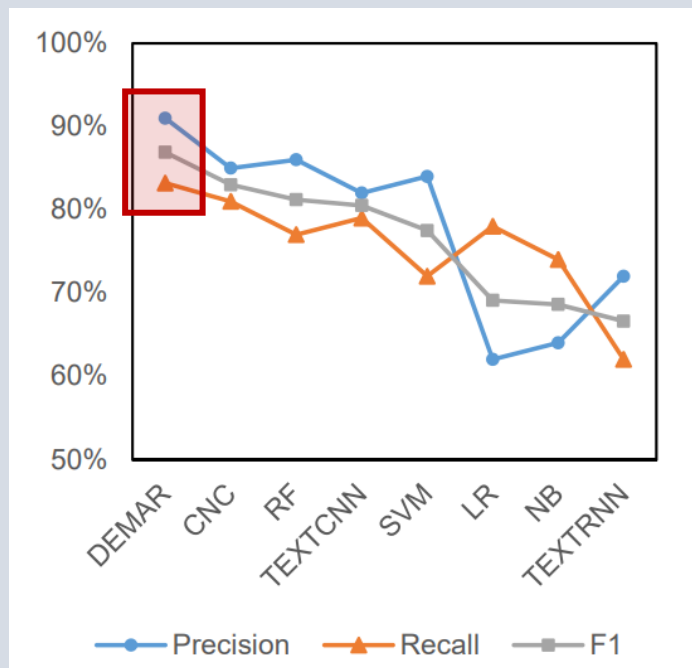
# RQ2：既存手法と比べて提案手法の性能が高いか？

## 方法

2つの既存手法(CNNベースの分類モデル(CNC)<sup>[18]</sup>とヒューリスティックなルールベースの分類モデル(FRA)<sup>[51]</sup>) + 6つの一般的なテキスト分類モデルと、提案手法を比較

## 結果

### ・ 要求発見



### ・ 要求注釈

| Approach | F1    |       |       |       |       |       |       | Accuracy |
|----------|-------|-------|-------|-------|-------|-------|-------|----------|
|          | INT   | BE    | DR    | EXP   | EXA   | TRI   | CB    |          |
| DEMAR    | 91.4% | 76.2% | 68.9% | 87.2% | 69.7% | 88.0% | 91.5% | 83.3%    |
| FRA      | 87.5% | 84.9% | 83.0% | 83.0% | 80.5% | 82.0% | NA    | 81.5%    |
| LR       | 70.5% | 48.6% | 56.8% | 57.4% | 71.0% | 79.2% | 77.4% | 68.6%    |
| NB       | 70.0% | 45.7% | 57.6% | 46.7% | 73.5% | 79.0% | 76.8% | 69.4%    |
| RF       | 75.9% | 64.4% | 72.5% | 54.0% | 60.4% | 80.0% | 78.5% | 72.0%    |
| SVM      | 75.8% | 59.4% | 69.4% | 57.4% | 71.8% | 74.1% | 77.9% | 75.7%    |
| TEXTCNN  | 74.9% | 51.5% | 61.7% | 67.2% | 54.9% | 83.8% | 78.0% | 73.2%    |
| TEXTRNN  | 57.7% | 38.0% | 58.5% | 61.0% | 53.4% | 82.0% | 47.3% | 58.3%    |

一部ヒューリスティックな方法に劣るが、提案手法は学習ベースなので、専門家の手作業でのルール構築に頼らずには済む

Answer：全体的には既存手法よりも分類性能向上

他のプロジェクトでも使えそうか？

機能要求の書き方は多くのプロジェクトで共通な部分が多いはず（例：“〇〇を実装して欲しい”，”××機能を改善すべきだ”）なので，多くのOSSプロジェクトで使える

文注釈は要求Issue以外にも使えそうか？

例えば，ほとんどのバグレポートでは，現状(CB)，欠点(DB)，説明(EXP)といった文が含まれているはずなので，要求Issueに限らず注釈を適用することで，バグトリアージなどのタスクでも注目したい文を抽出できる

## 目的

大量のIssueから要求発見/注釈の自動化

## アプローチ

要求発見/注釈タスクに 관련된 情報を，マルチタスク学習を用いてタスク間共有の  
特徴ベクトルに埋め込む

## 結果

- RQ1：シングルタスク学習と比べてマルチタスク学習は性能が高いか？

Answer：マルチタスク学習で分類性能が向上

- RQ2：既存手法と比べて提案手法の性能が高いか？

Answer：CNNベースやルールベースの既存モデルよりも分類性能が向上