

Prediction of Vulnerability Characteristics Based on Vulnerability Description and Prompt Learning

Xiangwei Li*, Xiaoning Ren*, Yinxing Xue*, Zhenchang Xing^{†‡}, Jiamou Sun[†]

*University of Science and Technology of China, Hefei, China

lixw20@mail.ustc.edu.cn, hnurxn@mail.ustc.edu.cn, yxxue@ustc.edu.cn

[†]CSIRO's Data61, Canberra, Australia

zhenchang.xing@anu.edu.au, u5871153@anu.edu.au

[‡]Australian National University, Canberra, Australia

Proceedings of the 30th International Conference on Software Analysis, Evolution and
Reengineering (SANER '23), pp604-615

発表者：松尾 和輝

背景

- ソフトウェアの脆弱性はセキュリティと信頼性に深刻な脅威をもたらすため早期に修正すべき
 - 脆弱性の報告数は莫大であるため、優先順位付けが必要
- Common Vulnerability Scoring System(CVSS)^[2]
 - 専門家の知識に基づき脆弱性の深刻度を決定するシステム
 - 影響度, 悪用可能性, 時間的, 環境的スコアを用いて評価
 - CVSSスコアの更新に**時間がかかる**^[7]

深刻度を予測するモデルの構築を目的とした研究が行われている

先行研究 | TextCNNやBiLSTMを用いた分類モデルを構築^[6, 7]

[2] FIRST, “Common Vulnerability Scoring System,” <https://www.first.org/cvss/>

[6] Z. Han, X. Li, Z. Xing, H. Liu, and Z. Feng, “Learning to predict severity of software vulnerability using only vulnerability description,” in 2017 IEEE International conference on software maintenance and evolution (ICSME). IEEE, 2017, pp. 125–136.

[7] X. Gong, Z. Xing, X. Li, Z. Feng, and Z. Han, “Joint prediction of multiple vulnerability characteristics through multi-task learning,” in 2019 24th International Conference on Engineering of Complex Computer Systems (ICECCS). IEEE, 2019, pp. 31–40.

先行研究の課題と本研究の目的

- 予測時に訓練段階で見たことのない単語に直面した場合、意味理解が不十分
- BERT^[12]やRoBERTa^[13]などの事前学習済み言語モデル (PLM) よりも性能が低い
- PLMを用いた深刻度分類モデルを構築すべき
 - 課題
 - pre-trainingとfine-tuningでタスクが異なるため、PLMの知識を十分に活用することができない
 - fine-tuningでは大量の訓練データが必要

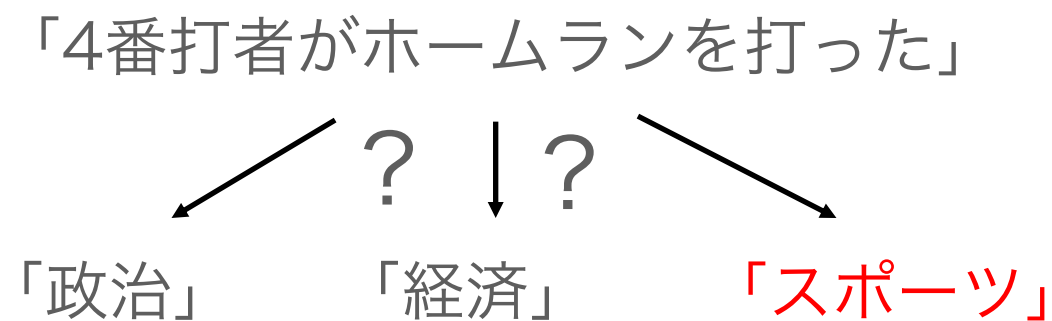
本研究の目的
上記課題を解決するための手法である **プロンプト学習** を用いて
深刻度分類モデル, 悪用可能性分類モデルを構築

[12] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, 2018.
[13] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," arXiv preprint arXiv:1907.11692, 2019.

プロンプト学習

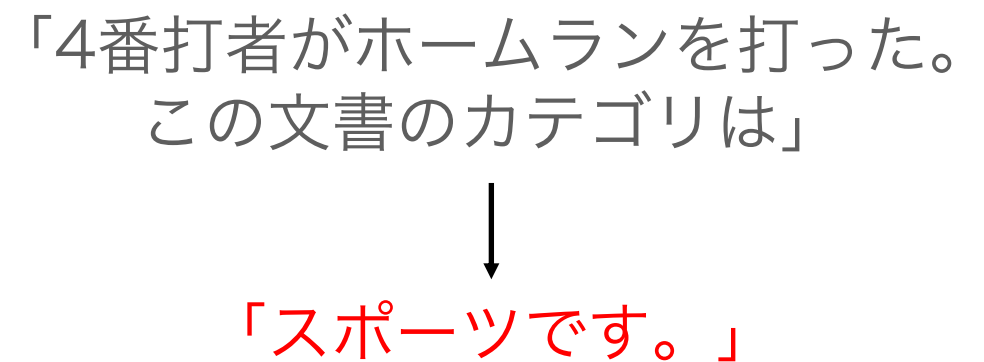
- 分類・NLIなどでラベルやカテゴリなどの形で定式化されているタスクを自然文生成の形にして解く手法

通常のカテゴリ問題



入力文に対してそのカテゴリを予測する分類器を学習し、新たな文に対して予測を行う

プロンプト学習



入力文に対してその次に来る単語を予測する言語モデルを学習し、新たな文に対して予測を行う

出典： OpenPromptを用いてpromptを試してみよう (<https://tech.retrieva.jp/entry/20220902>)

プロンプトテンプレートとVerbalizer

- プロンプトテンプレート
 - 入力スロット [X] と回答スロット [Z] の2つのスロットを持つテキスト文字列で構成
 - [X] には入力テキストが入り, [Z] にはPLMが予測した出力ラベルの単語が入る
- 例
 - テンプレート : "[X] The severity is [Z]."
 - 入力テキスト : "vulnerability description." の時
“vulnerability description. The severity is [Z].” になる
予測結果が “urgent” ならば
“vulnerability description. The severity is urgent.” になる

プロンプトテンプレートとVerbalizer

- Verbalizer

- 予測した単語をクラスラベルに変換するルール
- 赤枠で囲った単語が[Z]に出力されるように学習

$$\text{Verbalizer} = \left\{ \begin{array}{ll} \text{"Low"} : & [\text{"low"}, \text{"slight"}] \\ \text{"Medium"} : & [\text{"medium"}, \text{"moderate"}] \\ \text{"High"} : & [\text{"high"}, \text{"serious"}] \\ \text{"Critical"} : & [\text{"critical"}, \text{"urgent"}] \end{array} \right.$$

- 例

“vulnerability description. The severity is urgent.”ならば
urgent -> Criticalなのでこの文章はCriticalに分類

- プロンプトテンプレートとVerbalizerは手動で作成

プロンプトテンプレートの種類

- ハードプロンプトテンプレート

- 理解可能な自然語で構成されたプロンプト

$$f_{prompt} = \text{“The severity is [Z]. [X]”} \quad (1)$$

- ソフトプロンプトテンプレート

- ハードテンプレートに含まれる自然語を[SOFT]トークンに置き換えたプロンプト

$$f_{prompt} = \text{“[SOFT] [SOFT] [SOFT] [Z] [SOFT] [X]”} \quad (2)$$

- ハイブリッドプロンプトテンプレート

- タスクに関連する重要なトークンのみを残したプロンプト

$$f_{prompt} = \text{“[SOFT] severity [SOFT] [Z] [SOFT] [X]”} \quad (3)$$

リサーチクエスチョン

- RQ1：他の手法と比較して，プロンプト学習は脆弱性の深刻度分類にどの程度有効か？
- RQ2：サンプル数の少ない悪用可能性分類において，プロンプト学習はどの程度効果的か？
- RQ3：プロンプト設定の違いはプロンプト学習の性能にどのような影響を与えるか？

データセット

- NVD※から93,512件の脆弱性データを収集
 - CVSSの深刻度ラベルが付けられている
 - Low (2%) , Medium (40%) , High (43%) , Critical (15%)
 - (訓練, 検証, テスト) = (90%, 5%, 5%) に分割
- NVDから1,004件の悪用可能性データを収集
 - 「exploited」というラベルが付けられている502件のデータを収集
 - ラベルが付けられていないものをランダムに502件収集
 - (訓練, 検証, テスト) = (残り, 200件, 500件) に分割

ベースラインと評価指標

- ベースライン

- PLM, TextCNN, Attentionbased BiLSTMなど, いくつかの比較手法やモデルを利用
- PLM
 - BERT-base, BERT-large, RoBERTa-base, RoBERTa-largeを利用
 - fine-tuningとプロンプト学習を比較

- 評価指標

- 深刻度分類は重み付きF1値⁽⁸⁾, 悪用可能性分類はF1値⁽⁹⁾

$$w_i = \frac{\#sample_i}{\sum_{j=0}^3 \#sample_j}, \quad weighted-F1 = \sum_{i=0}^3 w_i * F1_i \quad (8)$$

$$F1-score = \frac{2 \cdot P \cdot R}{P + R} \quad (9)$$

RQ1 | 結果

RQ1：他の手法と比較して、プロンプト学習は脆弱性の深刻度分類にどの程度有効か？

- 従来手法よりもPLMの方が精度高
- fine-tuningよりもプロンプト学習を用いた方が精度高（有意差あり）

Model	Method	w -F1 (%)	p-value
TextCNN	-	74.95	-
Attention-Based BiLSTM	-	74.71	-
BERT-base	fine-tuning	78.73	5.96E-07
	prompt learning	80.45	
BERT-large	fine-tuning	79.62	1.62E-06
	prompt learning	81.16	
RoBERTa-base	fine-tuning	79.37	4.21E-06
	prompt learning	80.76	
RoBERTa-large	fine-tuning	79.79	2.15E-06
	prompt learning	81.29	

RQ1の回答
脆弱性の深刻度分類において、プロンプト学習はfine-tuningや他のモデルよりも良い結果（1~2%向上）を得ることが可能

RQ2 | 結果

RQ2：サンプル数の少ない悪用可能性分類において，プロンプト学習はどの程度効果的か？

- プロンプト学習の方がfine-tuningよりも2~10%精度高
- 転移学習についてもプロンプト学習の方が精度高

学習に使う訓練
データの数

model	method	0	32	64	128	256
RoBERTa-base	fine-tuning	50	53.62	62.67	70.56	74.13
	prompt learning	52.70	63.74	67.13	73.19	75.41
RoBERTa-large	fine-tuning	50	58.53	65.87	72.87	74.82
	prompt learning	53.13	64.24	68.89	74.55	76.75
transferred RoBERTa-large1	fine-tuning	50	62.31	67.61	74.13	75.84
transferred RoBERTa-large2	prompt learning	55.41	69.27	71.80	75.92	77.92

深刻度分類から
悪用可能性分類
への転移学習

RQ2の回答

プロンプト学習は，サンプル数の少ないシナリオに対応する能力が高い

RQ3 | 結果

RQ3：プロンプト設定の違いはプロンプト学習の性能にどのような影響を与えるか？

- Verbalizerを固定した時のテンプレート別の評価

	Template	Type	w-F1 (%)
前置	T1 The severity is [Z]. [X]	Hard	80.22
	T2 [SOFT]*3 [Z] [SOFT] [X]	Soft	80.34
	T3 [SOFT] severity [SOFT] [Z] [SOFT] [X]	Hybrid	80.39
	T4 The severity of the following vulnerability is [Z]. [X]	Hard	80.45
	T5 [SOFT] * 7 [Z] [SOFT] [X]	Soft	80.57
	T6 [SOFT] severity [SOFT]*3 vulnerability [SOFT] [Z] [SOFT] [X]	Hybrid	80.73
後置	T7 [X] The severity is [Z].	Hard	79.88
	T8 [X] [SOFT] * 3 [Z] [SOFT]	Soft	80.02
	T9 [X] [SOFT] severity [SOFT] [Z] [SOFT]	Hybrid	79.95

$$\text{Verbalizer} = \begin{cases} \text{"Low"} : & [\text{"low"}, \text{"slight"}] \\ \text{"Medium"} : & [\text{"medium"}, \text{"moderate"}] \\ \text{"High"} : & [\text{"high"}, \text{"serious"}] \\ \text{"Critical"} : & [\text{"critical"}, \text{"urgent"}] \end{cases}$$

- 全体的にHard < Soft <= Hybridの傾向
- 後置テンプレートよりも前置テンプレートの方が精度高
 - [Z]の位置が固定されており、文脈の意味をより効果的に抽出できるため

RQ3 | 結果

RQ3：プロンプト設定の違いはプロンプト学習の性能にどのような影響を与えるか？

- テンプレートを固定した時のVerbalizer別の評価

	Verbalizer	$w\text{-}F1$ (%)
V1	low: ["low"], midium: ["medium"] high: ["high"], critical: ["critical"]	80.21
V2	low: ["low", "slight"] midium: ["medium", "moderate"] high: ["high", "serious"] critical: ["critical", "urgent"]	80.73
V3	low: ["low", "slight", "small"] midium: ["medium", "moderate", "middle"] high: ["high", "serious", "big"] critical: ["critical", "urgent", "huge"]	80.32

T6 [SOFT] severity [SOFT]*3
vulnerability [SOFT] [Z] [SOFT] [X]

- Verbalizerの単語数が多ければ多いほど良いというわけではない
 - 追加された単語がタスクと無関係であれば、予測の妨げになるため

RQ3の回答

テンプレートとバーバライザーの設計はプロンプト学習の性能に影響を与える

まとめ

- 目的

- プロンプト学習を用いた深刻度分類モデル，悪用可能性分類モデルの構築

- 結果

- 脆弱性の深刻度分類において，プロンプト学習は finetuning や他のモデルよりも良い結果を得ることが可能
- プロンプト学習は，サンプル数の少ないシナリオに対応する能力が高い
- テンプレートとバーバライザーの設計はプロンプト学習の性能に影響を与える

所感

- Verbalizerの話がいまいちよくわからない
 - 赤枠で囲った単語以外のものが出力されたりしないの？
- 考察があまりない
 - プロンプト学習を試してみたら精度が上がりました!!!ってだけ