二段階のファインチューニングを行った BERT による変数定義抽出

山本 蒔志　加藤 祥太　加納 学
京都大学大学院情報学研究科
{shota, manabu}@human.sys.i.kyoto-u.ac.jp

概要

製造プロセスのデジタルツインを実現するためには，物理モデルが必要である．しかし，物理モデルの構築には多大な労力を要するため，我々はこの作業の自動化に取り組んでいる．本研究ではその要素技術として化学プロセス関連論文からの変数定義の抽出手法を提案する．提案手法は対象とする変数を特殊トークンに置換した文を BERT に入力することで文中の定義の位置を予測する．他分野のデータセットと化学プロセス関連論文データセットを順に用いて二段階のファインチューニングを行い，変数定義抽出モデルを構築した．提案手法を適用することで，沼本ら [1] の特徴量を用いた手法よりも高い性能を実現し，正解率 85.6%を達成した．

1　はじめに

化学や鉄鋼などの製造プロセスにおいて，物理モデルが活用されている．物理モデルは数式に基づいて設計され，実プラントでは計測できない状態量やプラントの将来の挙動を予測し，生産効率の改善や装置設計，制御系設計，運転条件の最適化などを行うのに活用される．多くの場合，物理モデル構築には，専門知識に加えて膨大な量の文献調査とモデル構築に必要な情報の抽出・統合が必要である．しかし，複数の文献の内容を精査しその関連性を把握するには，非常に多くの時間と労力がかかる．この負担を軽減するために，我々は複数の文献から情報を抽出し，組み合わせ，物理モデルを自動構築する人工知能（Automated physical model builder; AutoPMoB）の開発に取り組んでいる [2]．AutoPMoB を実現するには，文献に含まれる変数の定義を正確に抽出する手法が必要であり，本研究はこの手法の開発に取り組む．

本研究では定義抽出対象の変数を特殊トークンで置換した文を BERT（Bidirectional Encoder Representations from Transformers） [3] に入力して定義抽出を行う手法を提案する．提案手法はシンプルであり実装が容易であるだけでなく，すべての変数を統一的に扱うことができる．化学プロセスに関する論文より作成したデータセットを用いて，既存手法と提案手法の性能を比較する．

2　関連研究

変数の定義抽出にはいくつかの先行研究が存在し，変数の定義抽出のために複数の特徴量が提案されている．Lin らは変数と定義の距離，意味的な正しさ，品詞の文法的な妥当さの 3 つを特徴量として用い，複数の分類器をアンサンブルして定義を抽出する手法を提案した [4]．沼本らは Stanford Parser [5] を用いて定義の候補を抽出し，位置関係や定義らしさなどの特徴量をもとに，最も定義らしいと判断した候補を抽出する手法を提案し，F1-Score で 41.2% を達成した [1]．しかし，いずれの手法も AutoPMoB の要素技術として性能が不十分である．

近年，BERT に代表される事前学習モデルが多くの自然言語処理タスクで最高の性能を達成している．Kang らは SciBERT [6] によって文から専門用語及びその定義を抽出する手法を提案した [7]．彼らが定義抽出対象とした専門用語には変数も含まれるが，変数の定義抽出を行う場合，一般的な専門用語に比べ性能が低下することを報告している．

自然言語処理ワークショップ SemEval2022 にて，変数と定義の対応付けタスクである Symlink が提案された [8]．Symlink は，文中から変数と定義を抜き出す固有表現抽出タスクと，抜き出した名詞句の関連性を判別する関係抽出タスクの 2 つからなる．Symlink では Lee らが最も高い性能を達成した [9]．彼らの手法は文の先頭に疑問文を追加した単語列を SciBERT に入力して質疑応答を行うことで変数と定

義を抽出した．彼らは固有表現抽出では 47.61%, 関係抽出では 37.19%の F1-Score を達成した．

本研究は 2 つの点で Symlink とは問題設定が異なる．1 つ目は変数の抽出の有無である．Symlink では文中のどこに変数があるかは未知という状況を想定し，変数の抽出もタスクに含めていた．本研究で対象とする変数はすべて既知とするが，Lee [9] らは変数抽出タスクにおいて 99% 以上の recall を達成していたため，同じ手法でほぼ完璧に変数を抽出できると考えられる．2 つ目は抽出対象の種類である．Symlink では変数によって数量を指示された名詞句や，定義の内容を補足する名詞句などの定義以外の変数に関連する名詞句も予測の対象としていた．本研究は定義のみを対象とする．

3 データセット

3.1 化学プロセス関連論文データセット

化学プロセスに関連する論文計 45 報の変数に対して定義を付与したデータセット D_{Process} を作成した．ここで変数とは論文中に単独で現れる数学的記号のことであり，数式のみに現れる変数は対象外とする．D_{Process} は晶析プロセス [crystallization process; CRYST]，連続層型反応器 [continuous stirred tank reactor; CSTR]，バイオディーゼル生産プロセス [biodiesel production process; BD]，チョクラルスキープロセス [Czochralski process; CZ]，多管式熱交換器 [shell and tube heat exchanger; STHE] の 5 つのプロセスいずれかに関するものである．各プロセスの論文数と変数の総数を表 1 に示す．

3.2 Symlink データセット

定義抽出は固有表現抽出に近いタスクであるが，Devlin らが固有表現抽出を行ったデータセットには抽出の対象として約 20,000 の固有表現が含まれていたのに対して [3]，D_{Process} の抽出対象となる定義は約 1,000 であり，ファインチューニングに必要なサンプル数として十分でない．そこで，D_{Process} に追加で Symlink データセット (D_{Symlink}) に含まれる変数と定義の関係を用いる [8]．D_{Symlink} は情報科学，生物学，物理学，数学，経済学の 5 つの分野の合計 101 報の論文からなり，D_{Process} の約 10 倍の 16,642 個の変数を含む．

表 1 D_{Process} に含まれる論文数と変数の数

Dataset	Number of papers	Number of symbols
D_{CRYST}	11	410
D_{CSTR}	10	253
D_{BD}	9	327
D_{CZ}	8	380
D_{STHE}	7	421
D_{Process}	45	1,791

4 提案手法

4.1 BERT を用いた変数定義抽出手法

提案手法の概略図を図 1 に示す．まず定義抽出対象の変数を特殊トークン [target] で置換する．同じ変数が複数回登場する場合，そのすべてを [target] に置換する．次に置換した文を BERT に入力し，各トークンが定義の開始位置である確率と終了位置である確率を得る．開始位置が終了位置と同じか，より前にあるという条件のもと，開始位置と終了位置の確率の合計が最大となる箇所を探す．得られた開始位置から終了位置までの単語を定義として抽出する．

論文中の変数は必ずしも定義が明記されているとは限らない．例えば，慣習により使用法が決まっている π などの変数は大抵の場合定義が明記されていない．文脈から推測できる変数の定義も明記されないことがある．例えば，A_i について定義が与えられた後に，A_i に似た定義を有する変数 A_{i+1} が登場した場合，その定義は省略される．定義がない変数については，入力文の先頭 [CLS] トークンを抜き出すこととする．[CLS] トークンは定義にはなり得ないため，負例の正解として代用できる．

4.2 二段階ファインチューニング

定義抽出を行う場合に，D_{Symlink} 単独でファインチューニングを行っても,D_{Process} に含まれる定義を正確に抽出できるモデルを構築できないと予想される．変数定義は各分野ごとに多く用いられるパターンが存在するが，D_{Symlink} と D_{Process} は分野が異りそのパターンも異なるためである．また D_{Symlink} と D_{Process} でアノテータが異なるため，どこまで詳細な内容を定義に含めるかが一致しない．さらに D_{Process} のサンプル数はファインチューニングを行うのに十分でない．以上の問題点を解決するため，D_{Process} と D_{Symlink} の両方のデータセットを用いて二

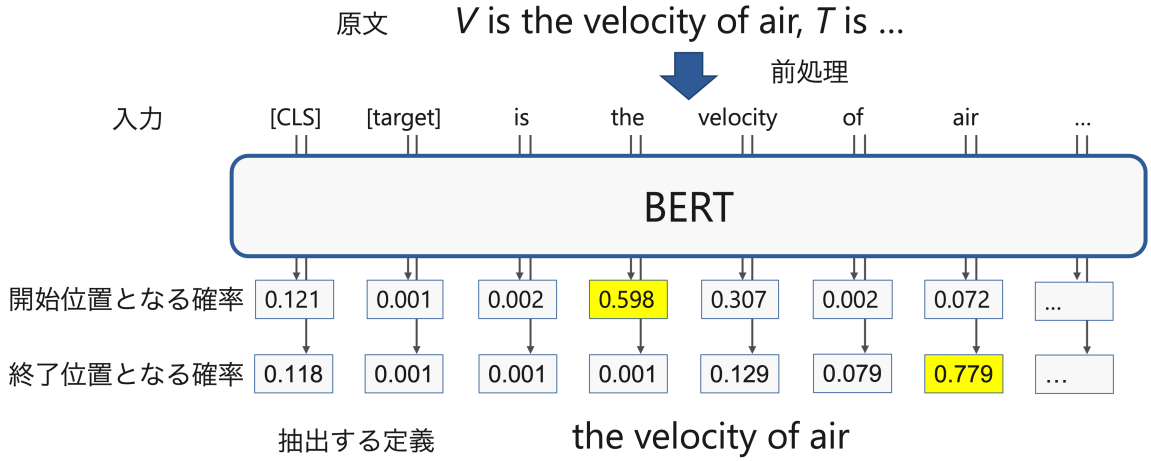

図 1　提案手法の概略図

段階のファインチューニングを行うこととした．まず第一段階のファインチューニングにて D_{Symlink} を用い，定義抽出タスクを行うモデルを作成する．次に第二段階のファインチューニングにて D_{Process} を用い，D_{Process} 中の定義のパターンにモデルを適応させる．

5　実験

5.1　前処理

D_{Process} と違い D_{Symlink} に含まれる文は TeX 形式であり，変数の表記に \$X\$, \$\mathcal{x}\$ などのコマンドを含む．このようなコマンドは，BERT が事前学習で用いた Wikipedia と BookCorpus の文に登場しないため，BERT のファインチューニングにおいてノイズとなる．そこで，pylatexenc [10] を用いて，TeX 形式の文をコマンドの含まれない Unicode 形式に変換した．

5.2　実験設定

D_{Symlink} を訓練用，検証用，テスト用に 8:1:1 に分割し，訓練用，検証用データのみをファインチューニングに用いた．ただし分割は各分野の文が等しい割合で含まれるように文単位で行った．また，D_{Process} を論文単位で訓練用，検証用，テスト用に分割した．沼本ら [1] と同じく D_{STHE} で 2 報，それ以外のデータセットで 3 報をテスト用とした．さらに各プロセスにおいて 1 報を検証用とし，残りを全て訓練用とした．

ベースモデルには DeBERTa-V3$_{\text{LARGE}}$ [11] を，オプティマイザには Adam [12] を，GPU には Google Colaboratory の Tesla T4 を用いた．バッチサイズは GPU のメモリで利用可能な範囲で最大の 8，学習係数は Symlink タスクにおいて最高の性能を達成したモデルと同じく 1e-5 を選択した．ファインチューニングは 5 エポック行い，各エポックにて検証用データに対するモデルの性能を確かめ，損失が最小となったモデルを性能評価に用いた．

二段階ファインチューニングが定義抽出の性能に与える影響を調べるため，提案手法に追加で以下に示す 2 つの方法で BERT のファインチューニングを行いそれぞれの性能を確認した．

1. D_{Symlink} のみでファインチューニング
2. D_{Process} のみでファインチューニング

5.3　評価方法

データセットのアノテーションされた定義に対して，モデルの予測した定義が完全に一致した場合を正解とする基準（full）と予測した定義が一部でも一致すれば正解とする基準（partial）の 2 つの評価方法にてモデルの評価を行った．また 1 つの変数に複数の定義が存在する場合は，アノテーションされた定義のうちのどれか 1 つを抜き出すことができれば正解とした．性能評価には D_{Process} のテスト用データを用いた．評価指標には正解率 (Acc.)，定義が存在する変数のうち正しく定義を抽出した変数の割合 (Rec.)，定義を抽出した変数のうち正しく定義を抽出した割合 (Pre.)，Pre. と Rec. の調和平均 (F1) の 4 つを用いた．実験は分割方法をランダムに変更しながら 10 回行い，その平均の性能を比較した．

表 2 従来手法と提案手法の性能及びファインチューニングに用いるデータセットを変更した場合の性能

Method	full				partial			
	Acc.	Rec.	Pre.	F1	Acc.	Rec.	Pre.	F1
Numoto et al. [1]	46.5	39.0	42.4	41.2	-	-	-	-
DeBERTa-V3 (D_{Symlink})	34.6	14.8	14.3	14.5	78.7	82.3	79.4	80.8
DeBERTa-V3 (D_{Process})	75.2	65.0	71.6	68.1	85.2	80.4	88.6	84.3
DeBERTa-V3 ($D_{\text{Symlink}} + D_{\text{Process}}$)	**85.6**	**81.4**	**81.7**	**81.5**	**90.6**	**89.8**	**90.2**	**90.0**

6 結果と考察

6.1 結果

提案手法，比較用の 2 つの方法，沼本らの手法による定義抽出結果を表 2 に示す．全ての評価指標について二段階ファインチューニングを行った場合が最高の性能となった．特に沼本らの従来手法に比べ，提案手法は各指標が 30 ポイント以上向上した．

6.2 二段階ファインチューニングの有効性

表 2 のように D_{Symlink} のみでファインチューニングを行ったモデルの full の性能が，他 2 つのモデルに比べて低くなった．主に 2 つの原因が考えられる．

1 つ目に，D_{Symlink} に含まれる論文と D_{Process} に含まれる論文は分野が異なるため，変数を定義する文のパターンが異なる．例えば，D_{Process} では “w_c is the mass flow rate of cold fluid (kg/sec).”のように，括弧の中に単位を記入するパターンが頻繁に登場する．一方で D_{Symlink} では，括弧が定義に含まれる例は 1 つもない．単位が定義に出現する頻度が D_{Symlink} では D_{Process} に比べて少ないなど，他にもいくつか定義に用いられるパターンに違いが見られるが，特に括弧の用法の違いは顕著であった．

2 つ目に D_{Symlink} と D_{Process} では定義の長さが異なる．図 2 に D_{Process} と D_{Symlink} の定義に含まれる単語数を示す．D_{Process} の定義は D_{Symlink} よりも長い傾向にある．D_{Symlink} では 1 単語または 2 単語の定義が 58%を占め，10 単語以上の定義は 3%に過ぎない．それに対して，D_{Process} では 1 単語または 2 単語の定義は 10%のみであり，10 単語以上の定義が 20%を占める．実際，D_{Symlink} のみでファインチューニングを行ったモデルは正解の定義よりも短い名詞句を抽出する傾向にあった．

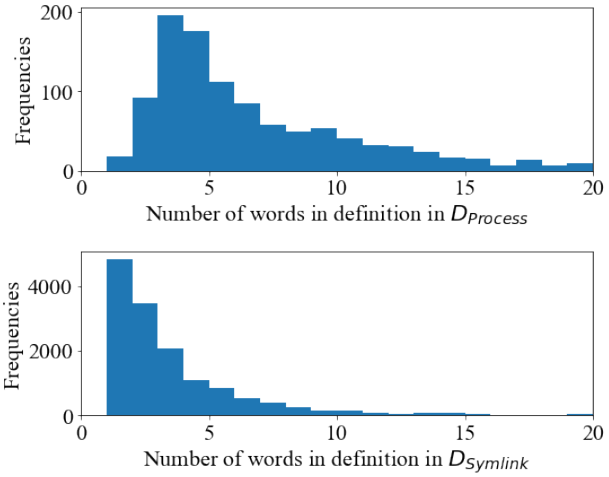

図 2 D_{Process} と D_{Symlink} の定義に含まれる単語数の比較

6.3 full と partial の性能差

D_{Symlink} のみでファインチューニングを行ったモデルについて，full の F1-Score は 14.5% であった一方で，partial の F1-Score で 80.8% となった．full と partial の性能差が大きいことから，一段階目のファインチューニングを行ったモデルは D_{Process} の定義の正確な位置は予測できないものの，D_{Symlink} と D_{Process} の定義に共通する大まかな位置を学習できたと考えられる．また，一段階目の大まかな学習により，D_{Process} のみでファインチューニングを行ったモデルよりも高い性能を達成できたと考えられる．

7 おわりに

本研究では対象とする変数を特殊トークンに置換した文を BERT に入力することで変数定義抽出を行う手法を提案した．さらに，化学プロセス関連のデータセットのサンプル数が不足するという問題を解決するため，二段階のファインチューニングによりモデルを構築した．従来手法と提案手法を比較した結果，提案手法は各指標において 30 ポイント以上従来手法を上回った．

謝辞

本研究は JSPS 科研費 JP21K18849 の助成を受けたものです.

参考文献

[1] 沼本真幸, 加藤祥太, 加納学. 変数の記号と定義に関する情報を活用した変数定義抽出手法. 言語処理学会年次大会発表論文集 (Web), Vol. 28th, pp. 486–491, 2022.

[2] Shota Kato and Manabu Kano. Towards an automated physical model builder: Cstr case study. In Yoshiyuki Yamashita and Manabu Kano, editors, **14th International Symposium on Process Systems Engineering**, Vol. 49 of **Computer Aided Chemical Engineering**, pp. 1669–1674. Elsevier, 2022.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)**, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[4] Jason Lin, Xing Wang, Zelun Wang, Donald Beyette, and Jyh-Charn Liu. Prediction of mathematical ex-pression declarations based on spatial, semantic, and syntactic analysis. In **Proceedings of the ACM Symposium on Document Engineering**, pp. 1–10.

[5] Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A python natural language processing toolkit for many human languages. **CoRR**, Vol. abs/2003.07082, , 2020.

[6] Iz Beltagy, Kyle Lo, and Arman Cohan. SciBERT: A pretrained language model for scientific text. In **Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)**, pp. 3615–3620, Hong Kong, China, November 2019. Association for Computational Linguistics.

[7] Dongyeop Kang, Andrew Head, Risham Sidhu, Kyle Lo, Daniel S Weld, and Marti A Hearst. Document-level definition detection in scholarly documents: Existing models, error analyses, and future directions. **arXiv preprint arXiv:2010.05129**, 2020.

[8] Viet Lai, Amir Pouran Ben Veyseh, Franck Dernoncourt, and Thien Nguyen. Symlink: A new dataset for scientific symbol-description linking. **arXiv preprint arXiv:2204.12070.**, 2022.

[9] Sung-Min Lee and Seung-Hoon Na. JBNU-CCLab at SemEval-2022 task 12: Machine reading comprehension and span pair classification for linking mathematical symbols to their descriptions. In **Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)**, pp. 1679–1686, Seattle, United States, July 2022. Association for Computational Linguistics.

[10] Philippe Faist. pylatexenc, 2021. https://github.com/phfaist/pylatexenc,version2.10.

[11] Pengcheng He, Jianfeng Gao, and Weizhu Chen. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing, 2021.

[12] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. **arXiv preprint arXiv:1412.6980**, 2014.