物理モデル自動構築に向けた変数アノテーションツールの開発

加藤 祥太　加納 学
京都大学大学院情報学研究科
katou.shouta.23v@st.kyoto-u.ac.jp　manabu@human.sys.i.kyoto-u.ac.jp

概要

製造業におけるデジタルツイン実現に不可欠な物理モデルの構築には多大な労力を要する．この問題を解決するために，著者らは，文献情報から物理モデルを自動で構築するシステム（Automated physical model builder; AutoPMoB）の開発を目指している．AutoPMoB の実現に向けた複数の要素技術の開発には専用のデータセットが必要である．本研究では，文献中の変数の記号とその定義を含むデータセットを作成するためのアノテーションツールを開発した．開発したツールを用いてプロセスの物理モデルに関する論文 28 報に含まれる変数の記号とその定義をまとめたデータセットを作成した．

1　はじめに

化学や鉄鋼などのプロセス産業において，物理や化学の法則に基づく物理モデルはプロセスの設計や運転に必要不可欠である．物理モデルの構築にはプロセスに関する深い理解と専門知識に加えて精度向上のための試行錯誤的な取り組みが必要なため，多大な労力を要するという問題がある．

著者らはこの問題を解決するために，文献情報から物理モデルを自動で構築する人工知能（Automated physical model builder; AutoPMoB）の開発を目指している．AutoPMoB は，1）文献データベースから対象プロセスに関する文書を収集し，2）文書の形式を HTML 形式に変換し，3）物理モデル構築に必要な情報（数式，変数，実験データなど）を抽出し，4）複数の文書から抽出した情報の同義性を判定して表現を統一し，5）その情報を統合して所望の物理モデルを構築する．

AutoPMoB を実現するためには上述の 5 つのタスクを実行するための要素技術を開発する必要がある．著者らはこれまでにタスク 3 に関連する変数を正確に抽出する手法の開発 [1] やタスク 4 に関連する数式群の同義性判定手法の開発に取り組んできた [2]．さらなる要素技術を開発するためにはプロセスに関する文献に情報を付与したデータセットが必要である．本稿では，変数の定義抽出と同義性判定の技術の開発に必要なデータセットを作成する．

従来の変数の定義抽出に関連する研究 [3, 4, 5, 6] は，Wikipedia の記事や arXiv.org に掲載されている論文を対象としており，HTML 文書中の `math` タグが付与された要素や TeX 文書中の数式をスペースで区切ることで得られるトークン，記述子などに対応する定義を抽出していた．プロセスに関する文献では，単一の記号で表される変数に加えて下付き文字がついた変数や複数の記号で表される変数などが登場するが，従来の研究ではプロセスに関する文献はほとんど扱われていない．また，従来の研究で用いることを想定したアノテーションツールがいくつか存在するが，工学分野の文献に特化したツールはなく，上述の変数と定義に特化したデータセットも存在しない．

そこで，本研究では，工学分野の文書から変数の記号を自動で抽出するアルゴリズムを提案し，抽出した変数の記号に対応する定義を付与する際に役立つアノテーションツールを開発する．さらに，そのツールを用いてプロセス関連の論文に対してアノテーションをおこなう．

2　変数を意味する記号の抽出方法

科学文書中の数式を対象とした研究の多くは HTML 形式または TeX 形式の文書を扱っている．数式は，HTML 形式の文書では Mathematical Markup Language（MathML）形式で，TeX 形式の文書ではコマンドを用いて記述されるため，それらから数式は容易に抽出できる．HTML 形式の文書が Web ページで入手可能であること，PDF 形式の文書を HTML 形式に変換するツールが開発されていることを踏まえ，本稿では HTML 形式の文書を対象とする．なお，LaTeXML [7] のようなソフトウェアで TeX 形式の文書を HTML 形式に変換することも可能である．

Algorithm 1 Variable symbol extraction

Input: d (HTML document)

Output: $\mathcal{V}$

1: $\mathcal{E}_{\mathrm{msup}} \leftarrow$ `msup` elements in d
2: **for** $i = 1$ to $|\mathcal{E}_{\mathrm{msup}}|$ **do**
3: $e_{\mathrm{msup},i} \leftarrow i$th element of $\mathcal{E}_{\mathrm{msup}}$
4: **if** superscript in $e_{\mathrm{msup},i}$ is not roman **then**
5: $s_{\mathrm{msup}} \leftarrow$ string in which exponent is represented using operator `^` instead of `msup`
6: replace the string matching $e_{\mathrm{msup},i}$ with s_{msup} in d
7: **end if**
8: **end for**
9: $\mathcal{E}_{\mathrm{math}} \leftarrow$ `math` elements in d
10: $\mathcal{V} \leftarrow \varnothing$
11: **for** $i = 1$ to $|\mathcal{E}_{\mathrm{math}}|$ **do**
12: $e_{\mathrm{math},i} \leftarrow i$th element of $\mathcal{E}_{\mathrm{math}}$
13: **if** $e_{\mathrm{math},i}$ has `mo` element **and** all texts included in `mo` elements are `&InvisibleTimes;` **then**
14: $s_{\mathrm{math}} \leftarrow$ string in $e_{\mathrm{math},i}$
15: add s_{math} to $\mathcal{V}$
16: remove s_{math} in d
17: **end if**
18: **end for**
19: $\mathcal{E}'_{\mathrm{math}} \leftarrow$ `math` elements in d
20: **for** $i = 1$ to $|\mathcal{E}'_{\mathrm{math}}|$ **do**
21: $e'_{\mathrm{math},i} \leftarrow i$th element of $\mathcal{E}'_{\mathrm{math}}$
22: $\mathcal{V}' \leftarrow$ variables in $e'_{\mathrm{math},i}$
23: $\mathcal{V} \leftarrow \mathcal{V} \cup \mathcal{V}'$
24: **end for**

プロセスの物理モデルに関する文献では，T のように単一の記号で表される変数だけでなく，T_{R} や $\hat{y}$，ΔE のように複数の記号で表される変数も存在する．このような変数を表す記号を自動で抽出するためのアルゴリズムを Algorithm 1 に示す．

この変数記号抽出アルゴリズムでは，まず，HTML 形式の文書 d に含まれる `msup` タグが付与された要素（上付き文字がついた記号）の一覧 $\mathcal{E}_{\mathrm{msup}}$ を取得する（1 行目）．各要素の上付き文字に該当する記号列がすべて立体（ローマン体）でない場合，その要素が累乗を表すと判断し，d に含まれる当該要素と同じ文字列を記号 `^` を用いた表記に置換する（2–8 行目）．これにより累乗と付加的な情報を表す上付き文字を区別する（例えば，T^{R} は 1 つの変数であるが，T^R は T の R 乗である）．次に，d から `math` タグが付与された要素の一覧 $\mathcal{E}_{\mathrm{math}}$ を得る（9 行目）．HTML 形式の文書における `math` タグが付与された要素は，TeX 形式の文書における数式環境（`\begin{equation}...\end{equation}`，`\begin{eqnarray}...\end{eqnarray}`，`$...$` など）に対応する．$\mathcal{E}_{\mathrm{math}}$ の各要素について，複数の記号が連続した文字列（ΔE や δt など）のみが存在する場合，その文字列を変数として抽出し，同じ文字列を d から削除する（11 行目から 18 行目）．最後に，上述の処理をした d から `math` タグが付与された要素の一覧 $\mathcal{E}'_{\mathrm{math}}$ を得て，各要素に含まれる変数を抽出する（19 行目から 24 行目）．ここで抽出される変数は 1 つの記号，または 1 つの記号に上付き文字，下付き文字，上下の装飾（$\hat{}$ など）のいずれか，あるいはそれらの組み合わせがついた文字列である．

3 アノテーションツール

科学文書中の変数を表す記号（変数記号）に対して定義（変数定義）を付与する際には，文書中の変数記号を特定し，変数定義に該当する名詞句を割り当てる．本稿ではこの作業の実行者をアノテータと呼ぶ．著者らは，アノテータが文書中の変数記号と変数定義をまとめたデータを作成するためのツールを開発した．

本ツールを用いる際，各変数記号に対して文中の変数定義と正しい変数定義を入力する必要がある．これは，文中の変数定義が変数記号の意味として必要十分ではない場合があるからである．例えば，C_{A} and C_{B} are concentrations of A and B. という文において，変数記号は C_{A} と C_{B} であり，それに対応する正しい変数定義は concentration of A と concentration of B であるが，文中の変数定義はいずれも concentrations of A and B である．文中の変数定義（concentrations of A and B）を抽出してから，それが必要十分かを識別し処理することで正しい定義（concentration of A と concentration of B）を得る定義抽出手法を今後開発していく予定である．

アノテーションツールの使用画面を図 1 に示す．最初に画面上部で対象とする文書を選択する（1）．本ツールは Algorithm 1 に従って選択された文書から変数記号を自動で抽出し ID を付与する．画面左側には文書中のテキストが表示される（2）．各変数記号が `MATH_ID` の文字列に置換されたテキスト（Processed text）と元のテキスト（Original text）の 2

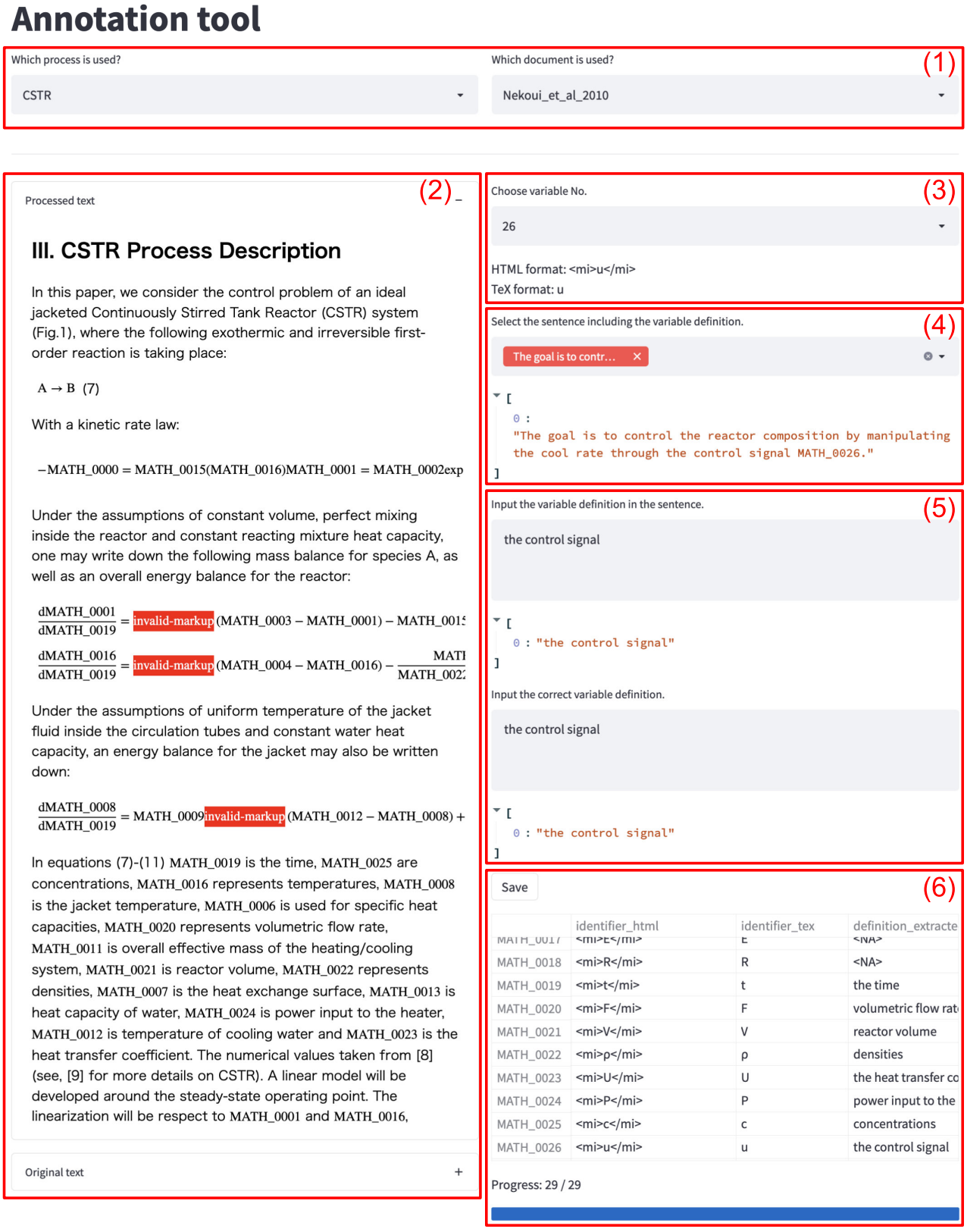

図 1　アノテーションツールの使用画面

種類を表示可能である．アノテーションは 1 変数記号単位でおこなう．アノテータは，画面右側でアノテーションをする変数記号を選択してから（3），変数定義を表す名詞句が含まれる文を選択し（4），

文中の変数定義と正しい変数定義を手作業で入力する（5）．複数の文に変数定義が登場する場合，それらをすべて抽出する．選択した文の数と入力した文中の変数定義の数が異なるときや選択した文に変数定義が含まれないときには警告文が出力される．入力が終了したら Save ボタンを押して情報をファイル（xlsx 形式）に保存する（6）．変数定義が存在する文がなければ文を選択せずに Save ボタンを押し，変数定義が存在する文がない，という情報を保存する．保存された情報は画面上で確認できる．

各変数記号に対して以下の 8 つの情報を保存する．

- ID
- 変数記号の文字列（HTML 形式）
- 変数記号の文字列（TeX 形式）
- 文中の変数定義
- 正しい変数定義
- 変数定義が存在する文の番号
- 変数定義が存在する文の文字列
- 正しい変数定義が抽出可能か否か（文中の変数定義と正しい変数定義が一致するか）

本アノテーションツールは Python で実装し，UI には Streamlit [8] を，テキストの文分割には Stanza [9] を使用した．

4 データセット作成

連続槽型反応器（CSTR）に関する論文 10 報，熱交換器に関する論文 7 報，晶析プロセスに関する論文 11 報を対象に，開発したツールを使用して変数記号と変数定義を含むデータセットを作成した．これらの論文は PDF 形式で入手したため，InftyReader [10] で TeX 形式に変換したのちに，もとの論文を再現するように人手で修正し，LaTeXML で修正後の TeX 形式の論文を HTML 形式に変換した．

作成したデータセットの統計情報を表 1 に示す．抽出された変数の総数は 1,094，文中に変数定義が存在する変数の数は 680，文中に存在する変数定義と正しい変数定義とが少なくとも 1 つ一致した変数の数は 513 であった．論文単位の変数の数の最小値は 16，最大値は 96，平均値は 39 であった．

このデータセットはアノテータ 1 名（第一著者）が作成した．作業ペースは論文に含まれる変数の数によるが，1 報あたり約 30 分であった．ツールを用いずに上述の 8 つの情報を手作業で保存する場合，論文 1 報あたりの処理時間は 1 時間以上だったため，本ツールによって大幅に作業を効率化できた．今後は，複数名のアノテータで上述の 3 つのプロセス以外のプロセスに関する論文と書籍を対象にしてアノテーションをおこない，データセットを拡張する予定である．

表 1 データセットの統計情報．V は変数集合，V_{ext} は文中に変数定義が存在する変数の集合，V_{cor} は文中の変数定義と正しい変数定義が一致した変数の集合である．$|V|$ は V に含まれる要素数を表す．

プロセス	論文数	$\lvert V\rvert$	$\lvert V_{\mathrm{ext}}\rvert$	$\lvert V_{\mathrm{cor}}\rvert$
CSTR	10	253	160	122
熱交換器	7	414	271	202
晶析	11	427	249	189
合計	28	1,094	680	513

5 おわりに

本研究では，製造プロセスの物理モデルに関する文献に含まれる変数を表す記号（変数記号）とその定義（変数定義）からなるデータセットを作成する作業を効率化するアノテーションツールを開発した．作業者（アノテータ）は本ツールによって自動で抽出される変数記号に対して，変数定義を手作業で入力する．開発したツールを用いて製造プロセス関連の論文 28 報に対して変数記号と変数定義を含むデータセットを作成した．本ツールとデータセットは公開予定である．

開発したツールは今後も改良を続けていく．具体的には以下の機能を追加予定である．

PDF 形式の文書の扱い 本稿では，PDF 形式の文書を TeX 形式に変換したが，数式や本文の一部が正しく変換されなかったため，手作業で修正した．今後，PDF 形式の文書を入力として扱えるように，TeX 形式への変換精度の改善に取り組む予定である．

定義の候補の提示 現状，アノテータは本文から定義に該当する名詞句を探し出す必要がある．アノテータがより効率的に定義を発見するためには，定義の候補を出力することが有効であると考えられる．アノテータの負担を軽減するために，変数定義の抽出技術を組み合わせ，定義候補を提示できるようにする予定である．

謝辞

本研究は JSPS 科研費 JP21K18849 および JST 次世代研究者挑戦的研究プログラム JPMJSP2110 の助成を受けたものです.

参考文献

[1] Shota Kato and Manabu Kano. Identifier information based variable extraction method from scientific papers for automatic physical model building. PSE Asia, Paper No. 210043, 2020.

[2] 張純朴, 加藤祥太, 金上和毅, 加納学. ルールベース手法による数式群の同義性判定. 言語処理学会第 27 回年次大会発表論文集, pp. 279–282, 2021.

[3] Giovanni Yoko Kristianto, Goran Topić, and Akiko Aizawa. Extracting textual descriptions of mathematical expressions in scientific papers. **D-Lib Magazine**, Vol. 20, No. 11, 2014.

[4] Moritz Schubotz, Alexey Grigorev, Marcus Leich, Howard S Cohl, Norman Meuschke, Bela Gipp, Abdou S Youssef, and Volker Markl. Semantification of identifiers in mathematics for better math information retrieval. In **SIGIR 2016 - Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval**, pp. 135–144, 2016.

[5] Moritz Schubotz, Leonard Krämer, Norman Meuschke, Felix Hamborg, and Bela Gipp. Evaluating and improving the extraction of mathematical identifier definitions. In **Conference and Labs of the Evaluation Forum (CLEF), Dublin, Ireland**, pp. 82–94, 2017.

[6] Hwiyeol Jo, Dongyeop Kang, Andrew Head, and Marti A Hearst. Modeling mathematical notation semantics in academic papers. In **Findings of the Association for Computational Linguistics: EMNLP 2021**, pp. 3102–3115. Association for Computational Linguistics, 2021.

[7] Bruce Miller. LaTeXML The Manual—A LaTeX to XML/HTML/MathML Converter, Version 0.8.3. https://dlmf.nist.gov/LaTeXML/, 2018. (Accessed on 2021/11/06).

[8] Streamlit • the fastest way to build and share data apps. https://streamlit.io/. (Accessed on 01/10/2022).

[9] Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A python natural language processing toolkit for many human languages. In **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations**, pp. 101–108, Online, July 2020. Association for Computational Linguistics.

[10] M. Suzuki, F. Tamari, R. Fukuda, S. Uchida, and T. Kanahori. Infty: An integrated ocr system for mathematical documents. In **Proceedings of the 2003 ACM Symposium on Document Engineering**, p. 95–104, 2003.