数式識別子の文書内曖昧性の解消

朝倉卓人
東京大学
takuto@is.s.u-tokyo.ac.jp

宮尾祐介
東京大学
yusuke@is.s.u-tokyo.ac.jp

概要

数式識別子の文書内曖昧性解消は、自然言語中の数式理解を実現する上で重要である。文書をまたいだ数式識別子の曖昧性解消については一定の進展が得られてきたが、文書内曖昧性は十分に研究されないまま残されてきた。本研究では、どのような情報が文書内曖昧性解消に必要であるのかを明らかにする。我々は文書内の位置データと数式識別子周辺の数式内ローカル構造が特に有効であると結論付けた。構築した多層パーセプトロンモデルは、人間アノテータに近い精度（一致率 85%、カッパ値 0.73）で文書内曖昧性解消を実現する。また、重要な情報種は対象文書の科学分野に依存しないことを確認した。

1 はじめに

科学技術文書の自動理解を達成する上で、数式の解析は避けて通れない。数式の性質は自然言語の性質とは大きく異なる。これらの重要な違いの 1 つは、単語の性質と数式識別子の性質の違いである。数式識別子には、多くの分野で慣習的に単一のアルファベットや記号、あるいは数文字程度の文字列が用いられ、説明的ではない。また数式識別子は、しばしば単一の文書内でも複数の意味で用いられるため、正しい数式理解には数式識別子の文書内曖昧性の解消が必須である。こうした問題は Presentation-to-Computable (P2C) 変換や数式情報抽出 (MathIR) などの応用タスクにおいて重大な問題として認識されてきた [11, 16, 20]。しかし、数式識別子の文書内曖昧性の分析に役立つデータ資源が不足していたため、これまでこの曖昧性について深くは知られてこなかった。そのため、どのような情報があればこうした文書内曖昧性の解消を行うことができるのかも不明であった。本研究では、独自に構築したデータセットを用いることで数式識別子の文書内曖昧性の自動解消に取り組み、どのような情報があればこの曖昧性解消を実現できるのかを明らかにする。

図 1 に文書内曖昧性の具体例を示す（図中の例文は PRML [6] から）。この例では、太字の $\mathbf{y}$ は機械学習アルゴリズムの挙動に対応する関数（概念 1）とその出力ベクトル（概念 2）の 2 つの意味で用いられている。我々の目的は、この例における第 1、第 3 の出現に概念 1 を、第 2 の出現に概念 2 を割り当てることである。同じ概念が割り当てられた出現同士は共参照関係にあるとみなせることから、このタスクは数式識別子に対する共参照解析とも言える。

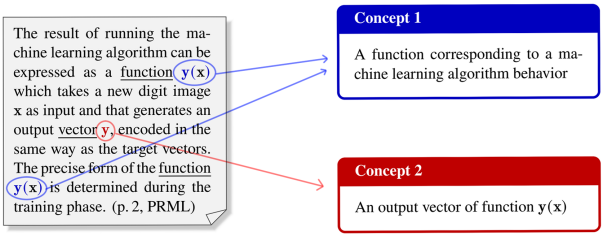

図 1 数式識別子の文書内曖昧性の例

2 関連研究

数式識別子などの数式内の小構造に対する解析タスクは、数学言語処理の中でも抽出タスクとして整理される。これらのタスクにはいくつかのバリエーションがあるが、説明アライメントと呼ばれるタスクが中心的である。このタスクは数式内の文字や記号（数式トークン）に対して、文書内外から対応する説明を見つけ出して付与するもので、Gaussian heuristic ranking [12] や K-means クラスタリングと SVM [14, 15]、Odin 文法に基づくルールベース手法 [2] を用いた自動化手法が提案されている。説明アライメント以外の抽出タスクとしては、Variable Typing [18] や Part-of-Math (POM) タグ付け [20] が提案されている。これらの多くのタスクと手法は、数式トークンの意味が単一の文書内において複数ある場合を考慮していない。

限定的な設定で数式トークンの文書内曖昧性を

考慮した研究は少数ある。Shan ら [16] は上付き添字、プライムなど数種類の数式構造について、それらの意味的な役割の分類を行う手法を提案した。Symlink [9] は複数のサブタスクから成る shared task で、このうち関係抽出タスクには単一段落内にある数式トークン間の共参照関係の特定が含まれる。この関係抽出タスクの SOTA 手法 [10] は F_1 スコア 37.19 と相対的に低く、大きな改善余地がある。

3 データセット

本研究では、我々が以前提案した数式グラウンディングデータセット[1)][5] を量的に拡張したものを利用した（表 1）。この拡張データセットでは、対象の arXiv 論文の中に現れるすべての数式識別子出現（計 27,657）に対して、対応する数学概念が手作業によりラベル付けされている。データセットに含まれる 40 本の論文のうち、半数の 20 本が NLP 分野の論文である。残りは 8 本が天文学、5 本が NLP 以外の情報科学、3 本が経済学、2 本が数学のものと、物理学と生物学のものが各 1 本ずつである。我々は NLP 分野の論文を MLP モデルの学習と評価の両方に利用し、それ以外の分野の論文は評価のみに利用した。また、NLP 論文のうち 4 本を最初に評価用データとして予約し、残り 16 本のデータを用いてモデル開発と訓練を行った。限られたデータを有効利用するため、モデルの開発時には原則として一つ抜き交差検証を利用した。

表 1 数式グラウンディングデータセットの拡張

データセット	論文	単語	識別子タイプ	出現	概念
オリジナル	15	86098	680	12352	1418
拡張版	**40**	237062	1742	**27657**	**3603**

4 タスク

数式識別子の文書内曖昧性解消を具体的な機械学習タスクとして定式化する。このタスクの目的は、各数式識別子出現に対して対応する数学概念を割り当てることである。タスクの入出力を次に示す。

入力：• 構造化文書データ（XHTML）
• 概念情報（各概念の紐付く初出位置）

出力：各出現に割り当てる概念

割り当て候補となる数学概念の情報は、各概念に紐付く最初の数式識別子出現の位置（初出位置）情報として与えられる。こうした初出位置では、しばしばその概念が定義されていることは注目に値する。初出位置情報はこのタスクにおける正解ラベルの一部に相当するが、数式識別子の出現数は数学概念の数よりも圧倒的に多い（表 1 参照）ため、与えられる正解ラベルは全体の 10%程度である。

1) https://sigmathling.kwarc.info/resources/grounding-dataset/ より入手可。今回の拡張データも後日同じリンク先より新バージョンとして提供予定。

5 多層パーセプトロンモデル

本研究では、多層パーセプトロン（MLP）モデルを用いて上記タスクを解決する。このタスクは、ラベルセット（数式識別子に割り当てる数学概念）が一定ではなく文書ごとに異なることから単純な多クラス分類問題とみなすことはできず、MLP の使用には工夫が必要となる。具体的には「割り当てを行いたい識別子出現の特徴」と「割り当て候補となる数学概念の特徴」のペアのベクトルを作成し、これを MLP の入力とする（図 2）。その上で、MLP はそのペアが正解である尤もらしさを出力するように訓練する。最終的に、出現・候補ペアのうち最も確率が高いものを選択することで、概念の割当を行う。

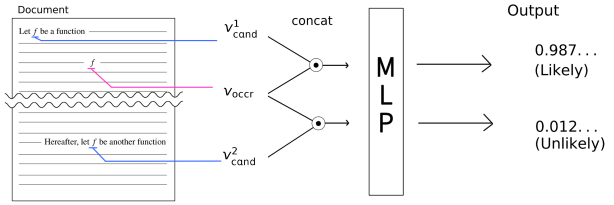

図 2 MLP アーキテクチャ

モデル性能の評価は著者らがデータセット構築時のアノテータ間一致率の算出に用いたのと同じ方法を用いる [5]。すなわち人間アノテータによって与えられた概念との完全一致率および Cohen のカッパ値 [8] を算出する。

モデルの学習にあたっては、平均二乗誤差を loss 関数として用いる。各種のハイパーパラメータは Optuna [1] を用いて探索した。探索の結果、オプティマイザには SGD、活性化関数には ReLU を採用した。隠れ層のサイズ、学習率、バッチサイズなどの重要度は相対的に低いが、いずれもモデルごとに機械的な探索により値を決定した（付録 A）。

6 特徴量エンジニアリング

MLP への入力を作成するにあたって、3 種類の情報を入力から抽出した（表 2）。第一の特徴は識別子出現周辺のテキスト文脈である。文脈を MLP に入力可能なベクトル表現に変換するにあたっては Sentence Transformer [13] を利用した。事前学習モデ

表 2 数式識別子出現から抽出された 3 種類の特徴

特徴	説明	例
文脈	出現前後のテキスト	`the feature vector $v'_{x}$ extracted from the`
接辞タイプ	数式内のローカル構造	プライムの有無、添字の有無、etc.
位置データ	Cascade 効果の有無と初出からの距離	Cascade 効果：あり、距離：20 単語

ルとしては MiniLM [19]、MPNet [17]、SciBERT [7] を比較し、本タスクにおいて最もよい性能を発揮した MiniLM をモデルの評価実験に利用した。文脈を特徴として利用する場合、本タスクは Semantic Textual Similarity (STS) タスクに類似するが、STS 用に fine-tuning されたモデルを用いても性能の向上は認められない。また文脈の window size や文脈内に現れる数式のエンコード方法（LaTeX または特殊タグ）は性能にほとんど影響を及ぼさない。

第二の特徴は接辞タイプと呼ばれる識別子周辺のローカルな数式構造である。例えば、ある出現が上付き添字やプライムなどの修飾子を持つかどうかといった情報が該当する。我々のデータセットには 26 種類の接辞タイプが人手でアノテーションされている。これらの情報は我々が作成したルールベースの検出器により、開発データにおいて精度 90.56% で判別することが可能である。表 3 にこの検出器がカバーする[2]すべての接辞タイプの検出精度を示す。この検出器のエラーの多くは以下のいずれかに分類される。

- パターンでは区別不能：例えば、上付き添字がべき乗の演算子かインデックスかはパターンのみでは見分けられない。
- アノテーションのエラー：カンマなどの一部の接辞タイプは見落とされやすい。

ただし接辞タイプ検出器の性能が数%程度向上しても、最終的なタスク性能にはごくわずかな影響しか与えない。そのため、検出器のルールを極端に複雑にしてまで精度向上を求める必要はない。

第三の特徴は位置データで、これには 2 種類の情報が含まれる。1 つは出現とペアになっている概念が最も直近で初登場したものかどうかを表す真偽値である。2 つめは出現の位置とペアの概念の初出位置の相対距離を $[-1, 1]$ にスケールした値である。データセットの分析結果から、数式識別子のスコープは Cascade 効果の影響を強く受けることが知られている [5]。すなわちある数式識別子出現の意味は、同じ識別子タイプの直近出現の意味と同一である可能性が最も高い。したがって、各概念の初出位置以外では一切スコープの切替が起こらないと仮定する単純なルールベース手法（Cascade ベースライン）は、機械学習モデルを使わなくてもそれなりによい精度となる（次節の図 3 を参照）。位置データの 1 つめの真偽値は、この Cascade ベースラインの出力そのものである。

2) この検出器は一部のルールベースのみで検出することがまったく不可能な接辞タイプ（2 種類）や、開発データの中には現れない接辞タイプ（5 種類）はサポートしていない。

表 3 接辞タイプの検出精度

接辞タイプ（例）	出現	再現率	適合率	F_1
subscript (B_a)	1914	0.971	0.898	0.933
superscript (B^a)	548	0.987	0.646	0.781
comma ($B(a_0, a_1)$)	150	0.940	0.613	0.742
semicolon ($B(a_0; a_1)$)	26	0.961	0.641	0.769
prime (B')	117	0.931	1.000	0.964
asterisk (B^*)	92	1.000	0.978	0.989
hat ($\hat{B}$)	93	0.924	0.934	0.929
tilde ($\tilde{B}$)	44	1.000	1.000	1.000
bar ($\bar{B}$)	146	0.863	0.984	0.919
over right arrow ($\overrightarrow{B}$)	4	1.000	1.000	1.000
over left arrow ($\overleftarrow{B}$)	4	1.000	1.000	1.000
open parenthesis ($B(a)$)	611	0.901	0.880	0.890
close parenthesis ($B(a)$)	611	0.901	0.880	0.890
open bracket ($B[a]$)	4	1.000	1.000	1.000
close bracket ($B[a]$)	4	1.000	1.000	1.000
open brace ($B\{a\}$)	5	1.000	0.454	0.625
close brace ($B\{a\}$)	5	1.000	0.454	0.625
vertical bar ($B(a_0 \mid a_1)$)	82	0.878	0.900	0.888
leftside base (a^B)	13	1.000	0.812	0.896

7 実験 I：モデル比較

本タスクの解決に重要な情報種を特定するため、先の 3 種類の特徴をさまざまな組み合わせで MLP モデルの入力とし、それらの性能を比較した。モデルは NLP 論文 16 本から成る開発データで訓練し、残り 4 本の NLP 論文を用いて評価した。20 個の異なるランダムシードで実験を行い、その平均を以ってモデルの評価とした。

比較のため、3 種類のベースラインを用意した。Random ベースラインはすべての割当をランダムに行う。Mode ベースラインは最初に登場した数学概

念をすべての出現に割り当てる。もう 1 つはすでに説明した Cascade ベースラインである。また、同じ評価データについて 2 名の人間アノテータが独立に作業した場合のアノテータ間一致率も算出した。

図 3 にモデル評価の結果を示す。まず今回比較を行った 3 つの特徴は、いずれも本タスクの解決に有効である。それぞれを単独で利用したモデルは、いずれも Random/Mode ベースラインの性能を上回る。今回の評価データには文書内曖昧性が一切ない識別子タイプも含まれるため、Random/Mode ベースラインであっても一致率はある程度の値になるが、カッパ値は 0 となることに留意されたい。

3 つの特徴の中では位置データが最も効果的であった。位置データ単独のモデルは Cascade ベースラインの性能とほぼ一致する。一方、残る 2 つの特徴は単独で使用した場合には Cascade ベースラインの性能を下回る。

続いて 2 つの特徴を組み合わせたモデルの評価に着目すると、位置データに接辞タイプを組み合わせたモデルは Cascade ベースラインの性能を上回り、今回の実験の中で最も精度が高い。一方、位置データに文脈を加えてもほとんど性能の向上は認められない。このことは文脈から得られる情報のほとんどが位置データに包含されることを示唆している。前節で述べたように、文脈中の数式表現を LaTeX 数式から特殊タグに置き換えても性能の劣化が見られないことから、文脈ベクトルには数式内部の情報がほとんど含まれていないこともうかがえる。一方、接辞タイプは数式内部の構造に関する情報をもつので、位置データがカバーしない情報を提供する。

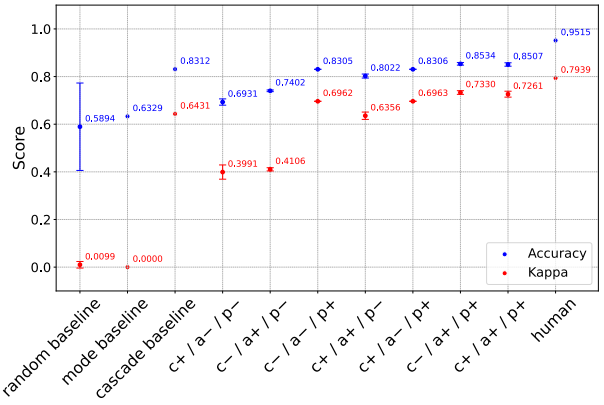

図 3　モデルの評価結果。モデルの名称は使用した特徴を示す。c は文脈、a は接辞タイプ、p は位置データで、使用の場合は +、不使用の場合は − が付される。例えば、c+/a+/p- は文脈と接辞タイプを利用するモデル。各点に付されたエラーバーは標準偏差。次の図も同じ。

3 つの特徴すべてを利用するモデルの性能は位置データと接辞タイプの組み合わせモデルとほとんど変わらない。

8　実験 II：分野依存性

前節の実験結果の分野依存性を確認するため、NLP 分野の論文で学習した MLP モデルを、非 NLP 分野の論文で評価した。3 節で述べたように、NLP 以外の各分野の論文本数にはばらつきがあるが、ここでは特定の分野の影響が大きくならないよう各分野の論文数が最大 3 となるように制限した。

結果を図 4 に示す。この図が示すモデル性能の傾向は、前節で述べたモデル性能の傾向と一致する[3)]。すなわち 3 種の特徴の中では位置データが最も有効である。位置データと接辞タイプの組み合わせが最高性能となる。位置データに文脈を加えても性能の向上は認められない。

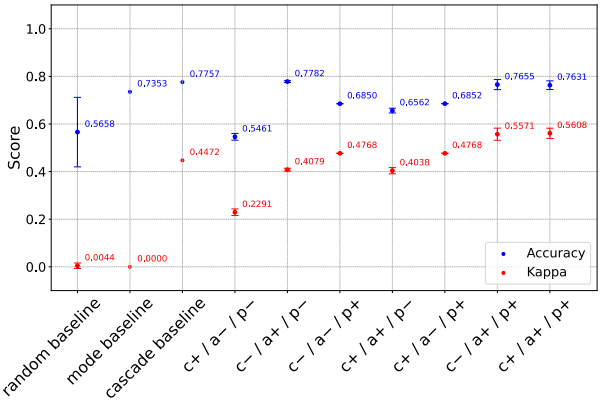

図 4　非 NLP データによる評価結果

9　結論

本研究では、ルールベース手法により抽出したさまざまな特徴量を利用して MLP モデルを訓練することで、どのような情報種が数学概念の割当タスクに有効かを特定した。具体的には、3 種類の特徴量を対象に比較実験を行い、その中で文書内の位置データが最も有効であることが明らかになった。次に有効な情報は接辞タイプと呼ばれる数式内のローカル構造である。数式前後の自然言語の文脈は 3 種類の特徴量の中では最も効果が限定的であった。これらの傾向は、対象論文の分野に依存しない。

3)　単純一致率は識別子タイプごとの割当難易度の差（候補概念が 2 つしかない場合もあれば、10 以上の場合もある）を考慮しないので、比較にはカッパ値を用いるのが妥当である。

謝辞

本研究は JST ACT-X (JPMJAX2002) の支援を受けて実施しました。

参考文献

[1] Takuya Akiba, et al. “Optuna: A next-generation hyperparameter optimization framework.” In *Proceedings of the* 25*th ACM SIGKDD international conference*. 2019.

[2] Maria Alexeeva, et al. “MathAlign: Linking Formula Identifiers to their Contextual Natural Language Descriptions.” In *Proceedings of the* 12*th Language Resources and Evaluation Conference* (LREC 2020).

[3] Takuto Asakura, André Greiner-Petter, Akiko Aizawa, Yusuke Miyao. “Towards Grounding of Formulae.” In *Proceedings of First Workshop on Scholarly Document Processing* (SDP 2020).

[4] Takuto Asakura, Yusuke Miyao, Akiko Aizawa, Michael Kohlhase. “MioGatto: A Math Identifier-oriented Grounding Annotation Tool.” In 13*th MathUI Workshop at* 14*th Conference on Intelligent Computer Mathematics* (MathUI 2021).

[5] Takuto Asakura, Yusuke Miyao, Akiko Aizawa. “Building Dataset for Grounding of Formulae — Annotating Coreference Relations Among Math Identifiers.” In *Proceedings of* 13*th Conference on Language Resources and Evaluation* (LREC2022).

[6] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[7] Iz Beltagy, Kyle Lo, 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* 9*th International Joint Conference on Natural Language Processing* (EMNLP-IJCNLP2019).

[8] Jacob Cohen. “A coefficient of agreement for nominal scales.” *Educational and Psychological Measurement*. 1960.

[9] Viet Lai, et al. “SemEval 2022 Task 12: Symlink — Linking Mathematical Symbols to their Descriptions.” In *Proceedings of Proceedings of the* 16*th International Workshop on Semantic Evaluation*.

[10] Sung-Min Lee, 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 Proceedings of the* 16*th International Workshop on Semantic Evaluation*.

[11] Jordan Meadows, André Freitas. “Introduction to Mathematical Language Processing: Informal Proofs, Word Problems, and Supporting Tasks.” *Transactions of the Association for Computational Linguistics* (2023).

[12] Robert Pagel and Moritz Schubotz. “Mathematical Language Processing Project.” In *Joint Proceedings of the MathUI, OpenMath and ThEdu Workshops and Work in Progress track at CICM* (2014).

[13] Nils Reimers and Iryna Gurevych. “Sentence-BERT: Sentence embeddings using siamese bert-networks.” arXiv:1908.10084 (2019).

[14] Moritz Schubotz, et al. “Semantification of Identifiers in Mathematics for Better Math Information Retrieval.” In *Proceedings of the* 39*th International ACM SIGIR Conference on Research and Development in Information Retrieval* (SIGIR2016).

[15] Moritz Schubotz, et al. “Evaluating and Improving the Extraction of Mathematical Identifier Definitions.” In *Experimental IR Meets Multilinguality, Multimodality, and Interaction* (CLEF2017).

[16] Ruocheng Shan, Abdou Youssef. “Towards Math Terms Disambiguation Using Machine Learning.” In *Intelligent Computer Mathematics* (CICM2021).

[17] Kaitao Song, et al. “MPNet: Masked and permuted pre-training for language understanding.” *Advances in Neural Information Processing Systems* 33 (2020).

[18] Yiannos Stathopoulos, et al. “Variable typing: Assigning meaning to variables in mathematical text.” In *Proceedings of the* 2018 *Conference of the North American Chapter of the Association for Computational Linguistics* (NAACL2018).

[19] Wenhui Wang, et al. “MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers.” *Advances in Neural Information Processing Systems* 33 (2020).

[20] Abdou Youssef. “Part-of-Math Tagging and Applications.” In *Intelligent Computer Mathematics* (CICM2017).

A　ハイパーパラメータ

モデルにより入力の次元数が異なることから、それぞれに Optuna [1] により最適化を行った。最終的に採用した値を次に示す。

表 4　各モデルのハイパーパラメータ

model	input size	hidden size	lr	batch size
c+ / a− / p−	768	694	0.639	23
c− / a+ / p−	52	47	0.328	70
c− / a− / p+	2	3	0.244	85
c+ / a+ / p−	820	701	0.310	21
c+ / a− / p+	770	335	0.483	57
c− / a+ / p+	54	34	0.698	35
c+ / a+ / p+	822	108	0.414	36