文と構造化情報のペアを同時生成する大規模言語モデルによるデータ拡張

加藤一翔 [1]　井田龍希 [1]　三輪誠 [1,2]
[1] 豊田工業大学　[2] 産業技術総合研究所人工知能研究センター
{sd22030, sd24501, makoto-miwa}@toyota-ti.ac.jp

概要

高性能な深層学習モデルに必要な学習データの確保のためにデータ拡張による自動生成が注目されている．しかし，構造化情報を扱うタスクでは，文と構造の対応付けが課題となりデータ拡張は未だ困難である．そこで本研究では，少数の例示から，大規模言語モデルを用いて文と構造化情報を同時に生成し，対応付けを不要とするデータ拡張手法を提案する．特に，生成データにおける語彙の偏りの軽減と生成データの整合性の検証による自己修正の工夫により，高品質なデータ生成を可能とする．生物医学分野のイベント抽出での検証の結果，提案手法で生成した事例による低資源下でのモデルの性能向上を確認し，提案手法の有効性を確認した．

1　はじめに

深層学習モデルは様々なタスクで高い性能を発揮しているが，その学習には大量のラベル付きデータが必要である．しかし，正確なラベル付けには専門知識と多大な時間を要するため，大量の学習データの確保は困難である．この解決策としてデータ拡張手法が広く研究されており [1]，近年は LLM を用いた手法が質問応答などで成果を上げている [2]．

しかし，情報抽出のように文と複雑な構造化情報のペアを扱うタスクでは，既存手法には限界がある．STAR [3] 等の逆生成アプローチは，構造化情報の生成後に，その構造化情報を条件に文を生成し，事後的に文字列マッチングを行う．この段階的な処理では図 1 のように条件にない構造化すべき情報が生成文に含まれる問題や構造化情報の要素の文からの欠落，同一語句の重複による対応付けの曖昧性など，文と情報間の不整合が問題となる．

そこで本研究では，LLM を用いて文と構造化情報をインライン形式で同時に生成することで，文と構造化情報の整合性を保持しつつ，事後的な対応づけを不要とする新たなデータ拡張手法を提案する．さらに，データの語彙の偏りを軽減する分布制御と生成データの整合性を検証・修正する自己修正の工夫により，高品質な学習データを構築する．

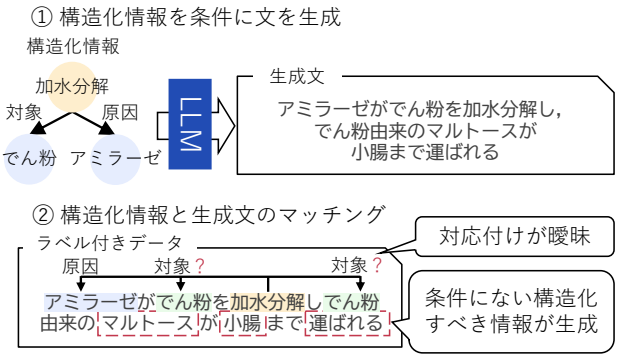

図 1　既存手法の課題

本研究の主な貢献は以下の通りである．

- 文と構造化情報の整合性を保持可能な同時生成によるデータ拡張手法を実現した．
- データの語彙の偏りを軽減する分布制御とデータの整合性を検証・修正する自己修正により，データ品質を改善した．
- 生物医学領域のイベント抽出を対象とした評価実験において，提案手法が低資源下でモデルの性能を向上させることを確認した．

2　関連研究

2.1　LLM を用いたデータ拡張

古典的なデータ拡張（EDA [4] 等）は文脈を破壊するリスクがあり，情報抽出への適用は限定的であった [5]．近年，LLM の高度な生成能力を基盤としたデータ拡張が注目され，二つに大別される．

汎用的な生成手法： DataGen [2] 等は文とラベルを生成するが，情報抽出に必要な文字単位の正確な位置

This work is published without peer review and is licensed by the author(s) under CC BY 4.0 (https://creativecommons.org/licenses/by/4.0/).

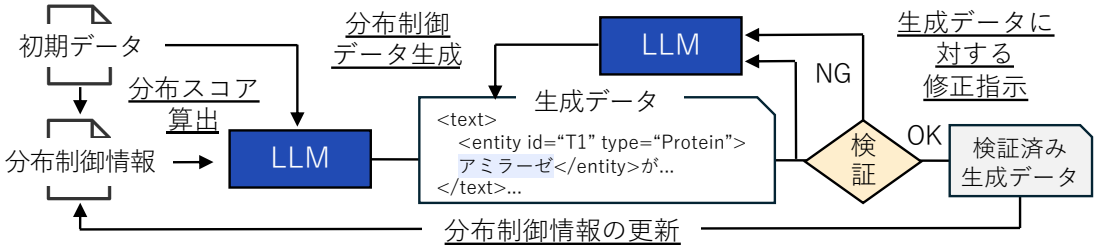

図 2　提案手法の構成：LLM による単一事例の生成と自己修正のフロー．生成のたびに分布制御情報を逐次更新し，このサイクルの繰り返しにより拡張データセットを構築する．

情報の出力が困難であり，位置ズレが頻発する．

逆生成：STAR [3] 等は構造を先に決定し文を生成する．構造の妥当性は担保されるが，文生成後の対応づけに課題が残る．

2.2　自己修正

自己修正とは，LLM の出力に含まれる誤りを，LLM 自身に修正させる手法である [6]．出力の構文エラーだけでなく，タスク固有の制約や内容を検証する点が特徴である．Adewumi ら [6] は，LLM に自身の出力内容を振り返らせる指示を与えることで，回答がどのように変化するかを検証し，推論を要するタスクにおける回答性能の向上を実現した．

3　提案手法

本研究では，文と構造化情報の不整合の問題を回避するため，インライン形式での同時生成による新たなデータ拡張手法を提案する．さらに生成データの自己修正機構と分布制御の工夫により，データの整合性を高め，語彙の偏りを抑えた高品質なデータ生成を可能にする．本手法はインライン形式に変換した初期データを元に LLM によるデータ生成および自己修正を用いたデータ拡張を行う．データ拡張の概要を図 2 に示す．以下，順に説明する．

3.1　データ整形：インライン形式の採用

情報抽出のデータセットでは，brat [7] や JSON 形式など，文書とアノテーション情報を個別に管理するスタンドオフ形式が一般的になっており，文字オフセットを用いてエンティティの位置を管理している．しかし，LLM はサブワード単位で処理するため，文字数の正確な扱いは難しい．この問題を回避するために，本研究では，LLM が扱いやすいインライン形式を採用し，文字数を考慮しない文と構造化情報の同時生成を可能とする．具体的には，インライン形式として XML 形式を採用する（図 3）．また，イベント情報は，その発生を示す語句（イベントトリガー）の ID を参照する形で記述する．

```
<document>
<text>
... A 17-year-old <entity id="T1000" type="Organism">girl</
      entity> ...
secondary to subretinal <entity id="T3" type="
      Blood_vessel_development">neovascularization</entity>
      ...
</text>
<events>
  <event id="E2" type="Blood_vessel_development" trigger="T3"/
      >
</events>
</document>
```

図 3　提案データ形式：構造化情報をインライン形式でテキストに埋め込むことでオフセット情報を不要にする．

3.2　データ生成：分布制御プロンプト

少数の初期データを例示として与え，生成すべきデータの指針と併せてプロンプトに含め，LLM によりデータを生成する．予備実験において，制約なしにデータ生成をした場合，例示したデータで頻度の高いエンティティ[1)]が多く生成され，頻度の低いエンティティが生成されにくい傾向が見られた．このエンティティの語彙的な偏りを防ぎ，データセット全体における語彙の多様性を確保するため，本手法ではスコアに基づく分布制御を導入する．

具体的には，各エンティティについて，初期データ内での比率（目標比率 P_{tgt}）と現在の生成済みデータ内での比率（生成比率 P_{cur}）から，目標比率からの乖離度を示す目標スコア $S = \frac{P_{cur} - P_{tgt}}{P_{tgt}}$ を算出する．S は，値が-1 に近いほど，そのエンティティの不足を意味する．なお，生成の初期は生成済みデータがないため，全スコアを −1 で初期化する．

本手法では，目標スコアの低い上位 50 件のエンティティを分布制御情報としてプロンプトに含める．この際，図 4 に示す通り，単語リストに加え，各エンティティの目標比率・生成比率および目標ス

1)　イベントの語彙にはイベントトリガーと固有表現があるが本稿では簡単のため，両者を併せてエンティティと呼ぶ．

This work is published without peer review and is licensed by the author(s) under CC BY 4.0 (https://creativecommons.org/licenses/by/4.0/).

```
### REFERENCE DISTRIBUTION
Prioritize items at the TOP (under-represented):
 * Cell|non-melanocytic cell: score=-1, target
     =0.19493%,current=0%
 * Planned_process|expressed: score=-0.77763, target
     =0.19493%, current=0.043346%
 ...
```

図 4 LLM に提示する分布制御情報の例．各エンティティの目標比率（target），現状の生成比率（current），および乖離度を示す目標スコア（score）を提示し，スコアが低い（不足している）ものを優先した生成を促す．

コアを数値としてプロンプトに含める．これにより，LLM に各語彙の不足度合いを考慮させ，生成プロセスを通して偏りを軽減し拡張データ群のエンティティ分布が目標分布へと収束するように促す．

3.3 自己修正：二段階検証と修正

生成データを XML およびドメイン制約の二段階で検証し，修正する．**XML スキーマ検証**では，タグの閉じ忘れや属性の欠落，ID の形式不正，未定義タグの混入といった XML の構文的な正しさを検証する．また，**ドメイン制約検証**では，データセットの定義に基づき，イベントトリガーの未使用，引数の型不一致，無効な参照，必須引数の欠落など，イベント構造の妥当性を検証する．これらの検証によって異常が検知された場合は，検知されたエラー内容に対応した事前定義された修正指示文をプロンプトに含めて再度 LLM に入力し，データの修正を促す．この修正は，設定した最大試行回数に達するか，異常が解消するまで繰り返し実行され，異常が検知されなかったデータを拡張データとして採用する．

4 実験

4.1 実験設定

分子から臓器レベルまでのイベントを定義した生物医学イベントコーパスである MLEE（Multi-Level Event Extraction）データセット [8] を用いて，提案手法を評価した．本コーパスは訓練 131 件，検証 44 件，評価 87 件の論文要旨から構成されている．本実験では，低資源状況を模倣するため，訓練データから無作為に抽出した 10 件のみを初期データとして使用した．性能評価は検証データを対象とし，Precision（P），Recall（R），F1 スコア（F）を用いた．データ生成には，Gemini 2.5 Pro [9] を使用し，生成データの修正は 5 回を上限とした．抽出には複雑なイベント構造抽出において SOTA を達成している BERT 基盤のイベント抽出モデルである DeepEventMine [10] を用いた．ベースラインとして，初期データ 10 件のみで DeepEventMine を学習させた**初期モデル**，および初期データ 10 件を例示して LLM に検証データの要旨から Few-shot 抽出を行わせた **LLM 直接抽出**を設定し，提案手法による拡張データを用いて学習したモデルの性能と比較した．

4.2 抽出性能の評価

ベースラインと提案手法による拡張データを用いて学習したモデルの比較を表 1 に示す．まず，LLM に初期データ 10 件を例示して直接抽出を行わせた結果，F1 スコアは 23.58%に留まり，学習モデルに及ばない結果となった．これより，LLM のコンテキスト内学習では既存テキストからの複雑なイベント構造の抽出は困難だとわかる．一方で，初期データ 10 件のみで学習した初期モデルと比較して，提案手法で生成した 50 件のデータのみで学習したモデルが F 値を 5.07%ポイント向上した．さらに，初期データに生成データを加えた場合は，F 値 30.43%と最も高い性能を記録した．この結果から，低資源下のイベント抽出においては，LLM を抽出器として直接用いるよりも，データ拡張を通じた抽出モデル学習の支援として用いる方が有効であるといえる．

4.3 分布制御のアブレーション解析

分布制御プロンプトの各構成要素が性能に与える影響を調査するため，拡張データ 50 件のみを用いたアブレーション解析を行った．比較対象として，インライン形式での同時生成は維持しつつ，提示する情報を制限した 4 つのパターンを評価した．結果を表 2 に示す．分布情報を用いない場合（None）と比較して，全情報を用いた提案手法は高い F 値を記録した．各要素を個別に検討すると，単語リストのみ（+用語）では性能が低下する傾向が見られた．一方，目標スコア（スコア）や目標比率・生成比率（比率）を加えることで，性能が向上した．これは，LLM が単なる初期データの模倣を超えて，データセット全体のエンティティ分布を考慮した生成をしたことを示している（分布の詳細は付録 A を参照）．

4.4 定性分析

本節では，提案手法が生成したデータの質を詳細に分析する．図 5 は，提案手法によって生成された

This work is published without peer review and is licensed by the author(s) under CC BY 4.0 (https://creativecommons.org/licenses/by/4.0/).

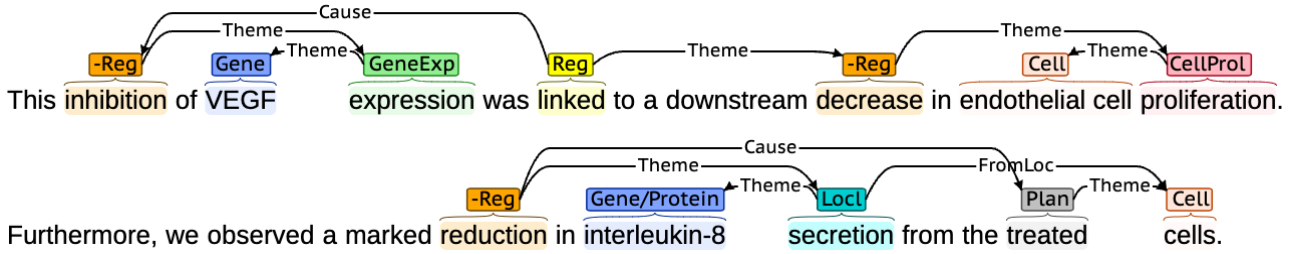

図 5　生成データの具体例．エンティティから複雑なイベントの入れ子構造までが整合的に生成されている．

表 1　抽出性能の比較（P/R/F1 は%表記）．DEM は DeepEventMine を指す．

手法（モデル, 学習データ）	P	R	F1
初期モデル (DEM，初期 10)	49.37	15.72	23.85
LLM 直接抽出 (Few-shot)	22.84	**24.37**	23.58
提案手法 (DEM，生成 50)	**53.70**	19.79	28.92
提案手法 (DEM，初期 10+生成 50)	53.20	21.31	**30.43**

表 2　分布制御プロンプトのアブレーション解析．拡張データ 50 件のみを使用し，P, R, F1 は (%) 表記．

プロンプト設定	P	R	F1
同時生成のみ（None）	52.64	17.07	25.78
＋用語	46.48	16.92	24.81
＋用語＋スコア	50.33	18.12	27.38
＋用語＋比率	**54.75**	17.56	26.59
提案手法 (全情報)	53.70	**19.79**	**28.92**

アノテーション例である．それぞれの例から，次の二つの高度な生成の特徴が確認できる．

一文目では，inhibition（Negative_regulation）が expression（Gene_expression）を対象とし，さらにそれらが linked（Regulation）を介して decrease（Negative_regulation）と結びつくといった事象間の複雑な依存関係の生成を確認できる．このようにインライン形式の同時生成により，イベントが別のイベントを引数として参照する階層の深い入れ子構造が，文の論理関係と整合性を保ったまま生成できている．

二文目では既知のエンティティを用いた高度な文脈拡充を確認できる．interleukin-8 が secretion や reduction（Negative_regulation）の対象として結び付けられ，さらに FromLoc を介して treated cells という介入条件まで構造に取り込まれている．このように，分布制御を用いても不自然にならずに，エンティティを適切に選んで，因果・条件付きの関係を含んだイベント構造を生成できている．

以上の分析から，提案手法は低資源下においても，複雑な事象間の論理構造と多様な専門表現を備えた高品質な学習データを生成できることがわかった．これは，表 1 に示した LLM による既存テキストからの抽出結果とは大きく異なった結果となっており，この差異の詳細な調査は今後の課題である．

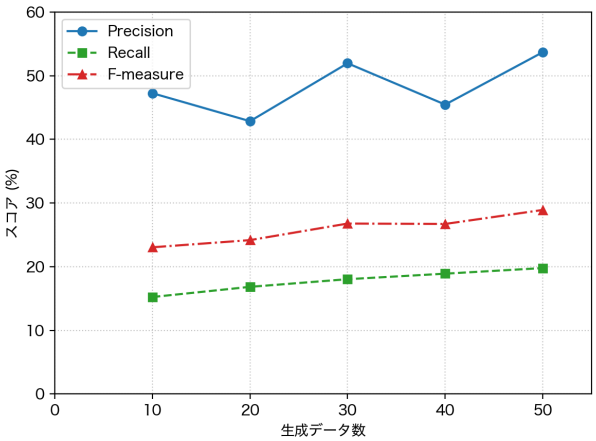

図 6　生成データ数と抽出性能の関係

4.5　データサイズと性能

生成数を 10 件から 50 件まで増やした際の影響を図 6 に示す．全体的にデータ増に伴い F 値は向上しており，学習に有効な新規パターンを生成できていることを示唆する．

5　おわりに

本研究では，情報抽出タスクにおける学習データの不足に対し，インライン形式で文と構造化情報を同時に生成するデータ拡張手法を提案した．本手法は，逆生成手法における対応づけの問題を回避し，データの語彙の偏りを軽減する分布制御と自己修正の工夫により品質を向上させた．生物医学領域のイベント抽出 (MLEE) を用いた評価実験では，提案手法で生成したデータにより，F 値を 5%ポイント以上改善し，多様かつ高品質なデータを生成できることを示した．今後の課題として，LLM のより詳細なデータ抽出とデータ拡張の振る舞いの違いの解析やより大きなデータや他のデータセットでの検証が挙げられる．

This work is published without peer review and is licensed by the author(s) under CC BY 4.0 (https://creativecommons.org/licenses/by/4.0/).

謝辞

この成果の一部は，NEDO（国立研究開発法人新エネルギー・産業技術総合開発機構）の委託業務（JPNP25006）の結果得られたものです.

参考文献

[1] Bosheng Ding, Chengwei Qin, Ruochen Zhao, Tianze Luo, Xinze Li, Guizhen Chen, Wenhan Xia, Junjie Hu, Anh Tuan Luu, and Shafiq Joty. Data augmentation using LLMs: Data perspectives, learning paradigms and challenges. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 1679–1705, 2024.

[2] Yue Huang, Siyuan Wu, Chujie Gao, Dongping Chen, Qihui Zhang, Yao Wan, Tianyi Zhou, Chaowei Xiao, Jianfeng Gao, Lichao Sun, and Xiangliang Zhang. DataGen: Unified synthetic dataset generation via large language models. In *Proceedings of the International Conference on Learning Representations*, 2025.

[3] Mingyu Derek Ma, Xiaoxuan Wang, Po-Nien Kung, P. Jeffrey Brantingham, Nanyun Peng, and Wei Wang. STAR: Boosting low-resource information extraction by structure-to-text data generation with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 18751–18759, 2024.

[4] Jason Wei and Kai Zou. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, pp. 6382–6388, 2019.

[5] Yang Zhou, et al. PGA-SciRE: Harnessing LLM on data augmentation for enhancing scientific relation extraction. In *Proceedings of the 23rd China National Conference on Computational Linguistics*, 2024.

[6] Hayato Tomisu, Junya Ueda, and Tsukasa Yamanaka. The cognitive mirror: A framework for AI-powered metacognition and self-regulated learning. *Frontiers in Education*, 2025.

[7] Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. brat: a web-based tool for NLP-assisted text annotation. In *Proceedings of the Demonstrations Session at EACL 2012*, 2012.

[8] Sampo Pyysalo, Tomoko Ohta, Makoto Miwa, Han-Cheol Cho, Jun'ichi Tsujii, and Sophia Ananiadou. Event extraction across multiple levels of biological organization. *Bioinformatics*, Vol. 28, No. 18, pp. i575–i581, 2012.

[9] Gemini Team, et al. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2024.

[10] Hai-Long Trieu, Thy Thy Tran, Khoa H. P. Duong, Anh Nguyen, Makoto Miwa, and Sophia Ananiadou. DeepEventMine: End-to-end neural nested event extraction from biomedical texts. *Bioinformatics*, Vol. 36, No. 19, pp. 4910–4917, 2020.

This work is published without peer review and is licensed by the author(s) under CC BY 4.0 (https://creativecommons.org/licenses/by/4.0/).

A　生成データのエンティティ分布

図 7 と図 8 は，初期データ 10 件において，出現頻度が高い上位 50 件のエンティティ（タイプ|名称の組み，例：Gene_or_gene_product|VEGF）を対象に，初期データ（黄色）と生成データ（青色）の出現割合を比較したものである．横軸は，各データ集合内における各エンティティの出現頻度を全エンティティの出現頻度で割った相対頻度を表す．

図 7 に示す通り，分布制御を行わない同時生成のみ（None）設定では，生成データの分布が特定の項目に極端に集中し，初期データにおける高頻度なエンティティを十分に再現できていない．これは，LLM が定型表現や頻出語彙を優先して出力する傾向に加え，一度偏った内容が生成されると，それに共起しやすい関連語彙も連鎖的に増えることが原因だと考えられる．

一方，図 8 に示す通り，提案手法（全情報）では，こうした分布の偏りが緩和され，多くの項目で黄色と青色の棒の長さが近づいており，生成分布が目標分布（初期データの分布）へと近づく傾向が確認できる．また，初期データでは出現頻度が低い語彙も一定割合で保持されており，これは 3.2 節で詳述した不足語彙の優先生成アルゴリズムが適切に機能した結果といえる．

このような分布の改善により，生成データが特定の語彙を過剰に生成することを防ぎ，多様な表現を含む高品質な学習事例の構築が可能となった．この語彙的多様性が，低資源下におけるモデル性能の向上（表 2）を支える重要な要因の一つであると考えられる．

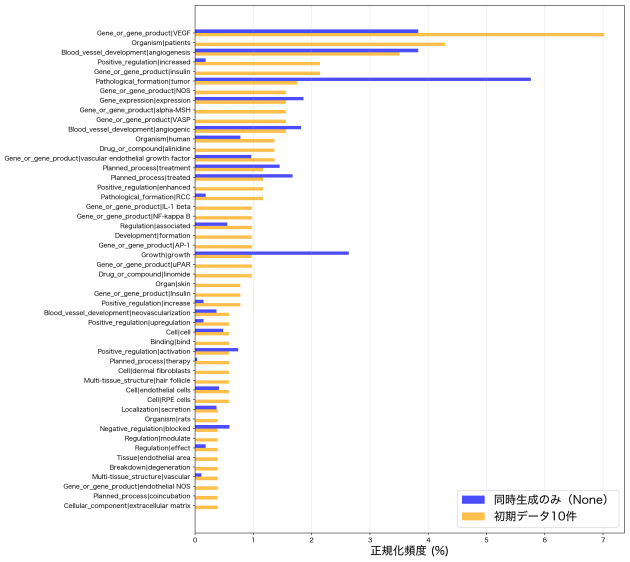

図 7　初期データと同時生成のみ（None）で生成したデータにおけるエンティティ分布の比較．縦軸は「タイプ|名称」（例：Gene_or_gene_product|VEGF）を上位から並べたもの，横軸は各データ集合内の全エンティティ出現数で正規化した相対頻度（%）を示す．

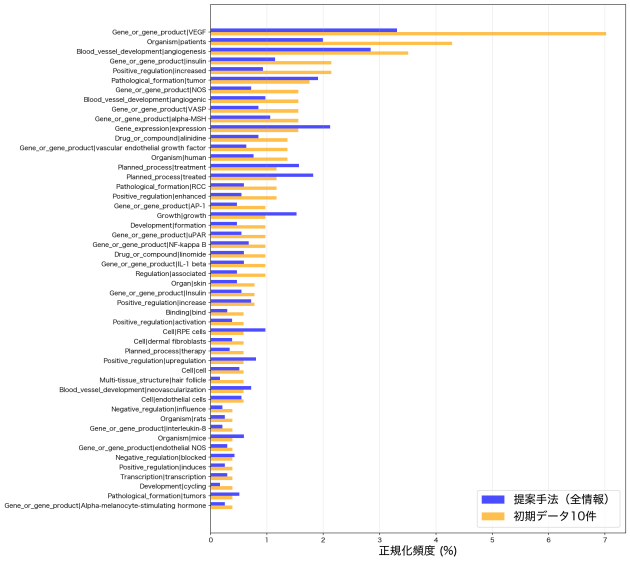

図 8　初期データと提案手法（全情報）で生成したデータにおけるエンティティ分布の比較．縦軸は「タイプ|名称」（例：Gene_or_gene_product|VEGF）を上位から並べたもの，横軸は各データ集合内の全エンティティ出現数で正規化した相対頻度（%）を示す．

This work is published without peer review and is licensed by the author(s) under CC BY 4.0 (https://creativecommons.org/licenses/by/4.0/).