生成 AI による化学文書への自動アノテーションとその評価

朱晨成 [1]　谷口友紀 [2]　大熊智子 [2]　嶋田和孝 [1]
[1] 九州工業大学大学院　[2] 旭化成株式会社
zhu.chencheng822@mail.kyutech.jp
{taniguchi.tcr,okuma.td}@om.asahi-kasei.co.jp　shimada@ai.kyutech.ac.jp

概要

材料・化学分野のテキストに対して，固有表現抽出を試みる場合，学習に使えるデータが十分でないことやアノテーションに高度な専門知識が必要であるという問題点がある．近年，隆盛を極めている生成 AI（GPT）を用いて固有表現抽出を試みることも可能だが，一般に，教師あり学習によりモデルをタスクに特化する方が良い精度が得られることが多い．そこで，本研究では GPT を用い，いくつかのプロンプトによる自動アノテーションを導入する．自動アノテーションによって得られた訓練データを基にラベル予測モデルを学習し，人間によるアノテーションデータとの精度差についても検証する．

1　はじめに

自然言語処理技術は材料・化学分野において大きな期待が寄せられている．特許文書や学術論文から材料に関する情報を抽出することで，新しい化学物を知識ベースに登録することができる．また，材料の合成プロセスを抽出することは開発の円滑化に寄与する [1].

材料に関する情報抽出には様々な課題がある．BERT に代表される事前学習済み言語モデルによって，様々な情報抽出タスクにおいて高い性能が示されている．しかし，一般に，それらのモデルの学習には大量の高品質な教師データが必要となる．公開されているコーパスは少なく，データ量の面で問題がある．さらに，化学文書へのアノテーションには専門的な知識が必要であるため，教師データの作成にコストがかかる．材料に関するコーパスも存在するが，特定のサブドメイン（例：燃料 [2]，電池 [3]）に焦点を当てるものが多い．また，それぞれが異なるガイドラインに基づいてアノテーションを行っていることが多く，機械学習のデータとしての汎用性にも問題がある．

近年，生成 AI を用いた自動アノテーションが注目されており，人手に迫る精度でアノテーションを行えることが報告されている [4]．その代表の 1 つとして，OpenAI が開発した大規模言語モデル GPT がある．GPT は特定のタスクに依存しない汎用的な言語モデルの構築を目的として作られた大規模言語モデル（Large Language Model: LLM）の一つである [5]．これまでにもテキスト要約，常識推論，感情分析など，様々な自然言語処理タスクで利用されている [6]．これらのタスクでは，GPT による自動アノテーションの有効性が示されている [7]．一方で，化学文書に対して生成 AI で自動アノテーションした事例はほとんどない．また，化学文書におけるデータ拡張による性能効果がの検証が行われていない．このような背景から，本研究では GPT を用いて化学文書に対して固有表現抽出（Named Entity Recognition: NER）のための自動アノテーションを行い，その性能を調査する．また，自動アノテーションによるデータを使ったラベル予測モデルも構築し，その性能を検証する．

2　関連研究

材料・化学分野における情報抽出　近年，材料・化学分野における情報抽出に関する研究では，テキストの特徴に焦点が当てられている．化学文書の特徴の 1 つとして，化合物の表記ゆれの問題だがある．化合物の部分構造や化合物特有の表記ゆれをそのまま扱うのは難しい．この問題に対処するため，Watanabe ら [8] は NER モデルと言い換えモデルによるマルチタスク学習を提案した．MatBERT [9] や MatSciBERT[10] のような材料科学に特化された言語モデルもある．これらのモデルは材料分野のタスクにおいて高い性能を示したが，専門性が高いデータセットを用いてファインチューニング行う必要

がある．Song ら [11] は，low-resource setting においてこれらのモデルの性能が劇的に下がることを示した．

GPT を用いた情報抽出　近年，代表的な LLM である GPT は様々な自然言語処理タスクにおいて高い性能を示している．しかし，NER タスクに利用する研究は少なく，そのほとんどが一般ドメイン（地名，人名，組織名など）で試みられている [6]，[12]．専門分野において，生物医学分野に焦点を当てている研究がある [13]，[14]．また，我々の知る限りでは，材料分野における GPT の情報抽出能力を検証した研究は 2 件しかない．Polak ら [15] は，一連の質問によるプロンプトを設計し，化学文書から材料，数値，単位を抽出した．実験では 90%以上の精度を得られたが，プロンプトのトークン長は長く，抽出されたエンティティは比較的に容易であった．Bölücü ら [16] は GPT-3.5 を用いて，より複雑な材料関連のエンティティを抽出し，few-shot 事例を選び出す適切な方法が回答の精度を向上させることを示した．

GPT を用いた自動アノテーションによるデータ拡張　これまでの研究において，GPT を情報抽出タスクに組み込んだ際，その性能は事前学習済み言語モデルより遥かに劣ることがわかっている．そこで，一部の研究では GPT をアノテーターとして導入している．GPT によってアノテーションされたデータは拡張データとして教師ありのモデルの学習に利用し，予測を行う．Wang ら [17] はラベルなしデータに GPT-3 を利用してアノテーションする方法について調査を行った．その結果，GPT-3 を用いた場合，人手アノテーションより 50%から 96%のコスト削減が可能であることが示された．しかし，彼らは生成タスクと感情分類タスクに焦点を当てており，本研究で対象とする NER のようなトークンレベルのタスクにおいての実験が行われていない．Ding ら [18] は GPT を用いた 3 つのアノテーションアプローチを設計し，AI ドメインにおける NER タスクでテストした．本研究では，より専門性が高い化学文書を対象とし，評価を行う．

3　対象データ

本研究では，手法の有効性評価に Materials Science Procedural Text Corpus (MSPT) [19] を用いる．MSPT は，230 個の材料の合成プロセスに関する英語の学術論文を含むデータセットであり，材料分野に関する知識を持つ専門家 3 名で人手アノテーションを行っている．材料，測定単位など 21 種のエンティティにラベルに付与されている．本実験では材料（MAT），操作（OPE），物性（PRO）の 3 つのエンティティを対象とする．

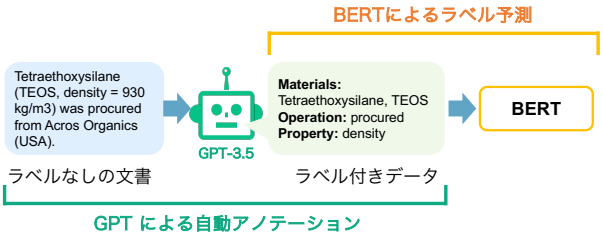

図 1　提案手法の概略．

4　提案手法

GPT を用いたアノテーションによるデータ拡張の性能を検証するため，本研究では 2 段階アプローチを提案する．図 1 に提案手法の概要を示す．緑色の部分が第 1 段階「GPT による自動アノテーション」であり，オレンジ色の部分が第 2 段階「BERT によるラベル予測」である．

4.1　GPT による自動アノテーション

GPT に関する多くの研究が適切なプロンプトを設計することの重要性を示している．そこで，第 1 段階ではプロンプトの内容について検討する．本論文では，（1）ロールの指定，（2）One-shot 正解例の有無について導入する．

（1）ロールの指定：Peng ら [20] は GPT にシステムロールを指定することで，特定タスクにおける性能が向上することを示した．本論文では，いくつかのパターンを検証し，「Material expert（材料分野の専門家）」と採用した．

（2）One-shot 正解例：正解例の導入には出力形式の維持や，直接的な参照を提供することで予測精度を向上させる効果がある．実験の際には，訓練データからランダムに 1 文を正解例として選択する．本論文では，*a*) one-shot 有，*b*) one-shot 無の 2 つのケースを比較する．

上記以外にもプロンプトにはタスクの説明や対象となるエンティティの説明，出力形式などについても記述する．プロンプトの全体像は Appendix A に示す．

また，GPT によるアノテーション方法について

も検討する．今回対象としたエンティティタイプ（MAT，OPE，PRO）はそれぞれに一定の関係性がある．したがって，あるエンティティタイプを推定する際に，別のエンティティタイプの情報も考慮する方が GPT は推定をしやすいかもしれない．つまり，3 つのエンティティタイプを同時にアノテーションすることがよい影響を与える可能性がある．これを「同時アノテーション」と呼ぶ．一方で，3 つの要素を同時に推定するよりは，各エンティティタイプを単独で二値分類するほうが問題設定としてはシンプルであり，精度が向上する可能性がある（たとえば，MAT かそうでないか）．これを「単体アノテーション」と呼ぶ．前述のプロンプトとアノテーション方法を組み合わせ，その有効性を検証する．

4.2 BERT によるラベル予測

前節で自動アノテーションされたデータを踏まえ，事前学習モデル BERT をファインチューニングし，3 つのエンティティタイプについて固有表現抽出を行う．その際，3 つの適用方法が考えられる．1 つめは，BERT が学習に利用するデータはすべて前節での自動アノテーション結果とする場合である．これを All GPT と呼ぶ．2 つめは，GPT が自動アノテーションした結果と人手によるアノテーションのデータを適当な比率で混ぜ合わせて学習する場合である．これを Mix と呼ぶ．3 つめは，すべてが人手によるアノテーションデータを用いる場合である．これを Human と呼ぶ．

Human に関しては，すべてのデータを学習に利用する Human_{Upper} と，一部のデータだけを用いる Human_{Low} の 2 種類を考える．Human_{Upper} による BERT モデルは，本論文での上限値にあたる．Human_{Low} は訓練データが少量しかない場合を想定した実験設定にあたる．Human_{Low} と All GPT や Mix との比較により，GPT による自動アノテーションによるデータ拡張の有効性を検証する．

5 実験

5.1 GPT による自動アノテーションの精度

プロンプトでの各エンティティの対応例および one-shot 正解例は，MSPT の 1 文書から選ぶ．その 1 文書を除いて，MSPT から 15 文書を選びテストセットとする．評価指標として各エンティティには Precision，Recall と F1 スコアを使用する．アノテーション方法はすべてのエンティティを同時アノテーションする場合，ミクロ平均を全体の評価指標（Overall）とする[1)]．GPT の実装について，モデルは gpt-3.5-turbo-0613[2)]，Temperature は 0 に設定する．

実験結果を表 1 に示す．なお，実際にはこれ以外のプロンプトについても検証している．詳細は Appendix B を参照のこと．太字は各エンティティでの各評価指標の最高値を意味する．表より，one-shot の導入により精度の向上が見られた．特に各エンティティの Precision が上がった．アノテーション方法については，MAT においては単体アノテーションが良い結果を得た．OPE と PRO については逆に同時アノテーションが効果的であった．これより，エンティティタイプごとに最適なアノテーション方法があることがわかる．

5.2 BERT によるラベル予測の精度

実験では MSPT のデータ分割方針に則り，訓練データには 200 文書，検証データには 15 文書，テストデータには 15 文書を用いる．5.1 節の結果を受け，それぞれで最適な手法（MAT：単体，OPE と PRO：同時・One-shot 有）を用い，訓練データに自動アノテーションを行ったものを 4.2 節の All GPT や Mix として扱う．検証データについては，人手によるデータをそのまま使う場合を H とし，自動アノテーションした結果を利用する場合を G とする．評価指標として各エンティティには F1 スコアを使用し，全体にはミクロ平均 F1 スコアを使用する．BERT は，BERT_{BASE}（*bert-base-uncased*）[3)] を用いる．BERT のファインチューニングについて，最適化アルゴリズムに AdamW，学習率は 2e-5 とし，損失関数には CrossEntropy を用いる．Epoch 数は 20，過学習抑制のために EarlyStopping を用いる．

Human_{Low} では，人手データが r = 5%，10%，20% の 3 パターンを試す[4)]．また，Mix においても，同様に 5%，10%，20% の 3 パターンを試す．Human_{Low} で 5%とは，訓練データの 5%のみを使ってファインチューニングすることを意味し，Mix での 5%とは，人手の訓練データが 5%で，残りの 95%は GPT により自動アノテーションされたもの用いることを意味

1） 単体アノテーションは，それぞれのエンティティタイプごとに評価をする．その結果，同時アノテーションと同じ基準でミクロ平均を求めることができないため記載していない．
2） https://platform.openai.com/docs/models/gpt-3-5
3） https://huggingface.co/bert-base-uncased
4） 結果の頑健性を確保するため，同じ割合でランダムにデータを抽出し，3 回実験をした平均値を結果とする．

表 1 異なるプロンプトにおけるアノテーションの精度．括弧内はテストデータでの事例数．

Entity (Support)		同時アノテーション						単体アノテーション		
		One-shot 無			One-shot 有			One-shot 有		
		Pre.	Rec.	F1	Pre.	Rec.	F1	Pre.	Rec.	F1
MAT	(360)	0.45	0.48	0.47	0.53	0.61	0.57	**0.61**	**0.62**	**0.62**
OPE	(254)	0.56	0.72	0.63	**0.69**	0.81	**0.77**	0.63	**0.88**	0.74
PRO	(103)	0.06	0.17	0.09	**0.12**	**0.17**	**0.14**	0.06	**0.49**	0.10
Overall	(717)	0.37	0.52	0.43	0.52	0.64	0.57		-	

表 2 3 つのアノテーションアプローチにおける BERT_{BASE} の予測精度．

		Human	**Human**		
			r = 5%	*r* = 10%	*r* = 20%
MAT	(652)	0.83	0.39	0.55	0.72
OPE	(285)	0.79	0.01	0.71	0.80
PRO	(192)	0.54	0.00	0.30	0.50
Overall	(1129)	0.76	0.30	0.56	0.70

		All GPT		**Mix**					
				r = 5%		*r* = 10%		*r* = 20%	
検証データ		*G*	*H*	*G*	*H*	*G*	*H*	*G*	*H*
MAT	(652)	0.64	0.54	0.64	0.62	0.66	0.68	0.66	0.67
OPE	(285)	0.72	0.69	0.75	0.73	0.72	0.72	0.77	0.75
PRO	(192)	0.10	0.03	0.12	0.07	0.11	0.13	0.19	0.16
Overall	(1129)	0.56	0.53	0.58	0.58	0.59	0.60	0.61	0.62

する．なお，Human_{Low} と Mix ではどのデータを選ぶかでランダム性があるため，評価の際は 3 回実験を行い，その平均を取る．

表 2 に 4.2 節で説明した BERT モデルの予測精度を示す．Human_{Upper} と比較すると，All GPT の精度は当然ながら低いが，OPE では Human_{Upper} と All GPT の差は小さい（0.79 vs. 0.72）．さらに，Human_{Low} における r = 5% と Mix における r = 5%（つまり 95%を GPT で補う）を比較すると，その差は歴然である（Overall で 0.30 vs. 0.58）．また，テストデータが異なるため単純に比較はできないが，Mix で（r = 10%, H）の場合の Overall の値は，表 1 の同時・One-shot 有の場合と比較しても高く（0.60 vs. 0.57），自動アノテーションよるデータ拡張とそれを用いたモデルの学習の有効性を示している．一方で，人手による訓練データが 20%程度ある場合は，Mix は Human_{Low} に勝てず，訓練データの拡張という点での GPT による自動アノテーションの限界も垣間見えた．これは，対象が材料・化学系分野であるという問題の難しさが原因であると考えられる．また，GPT は汎用言語モデルであり，材料・化学分野という特殊なテキストとの特性の違いも影響してるだろう．自動アノテーションの結果がノイズとなり，モデルの予測性能に悪い影響を及ぼしたと考えられる．

興味深い点としては，検証データの種類による精度差が挙げられる．All GPT では検証データに正解データである H よりも誤りも含む自動アノテーションの結果である G を用いる方がすべてのエンティティで精度が高い．Mix においては検証データが G であるか H であるかで大きな差は生じていない．機械学習を用いてモデルを作成する場合，訓練データ以外にもパラメータをチューニングする検証データが不可欠である．実験結果から，正確さが求められる訓練データには GPT による自動アノテーションにはまだ多くの問題があるが，パラメータチューニングに利用する検証データとしては有効に機能することが示されており，自動アノテーションによるデータ拡張はコスト面では有意義であるといえる．

6 まとめ

本論文では材料・化学分野におけるテキストデータからの情報抽出のために，生成 AI（GPT）を用いた自動アノテーションによるデータの自動拡張を試みた．プロンプトとアノテーション方法について検討し，GPT による自動アノテーションの精度を比較した．実験結果から，対象とするエンティティタイプによって適切なプロンプトを設計する必要があることがわかった．次に，この自動アノテーションデータを利用した予測モデルの構築を行った．Human_{Low} と Mix の比較により，極めて訓練データが少ない場合には自動アノテーションが有効であることが実験結果から示された．また，訓練データではなく，機械学習の検証データとして自動アノテーションが人手アノテーションと同等に機能することも分かった．

今後について，現在の自動アノテーションでは One-Shot を用いたが，より高い精度を実現するために few-shot の導入と，few-shot での適切な事例の選び方について考察する必要がある．

参考文献

[1] 有馬隆広, 大熊智子, 出羽達也. 新規用途探索を目的とした技術文書からの材料情報抽出. 言語処理学会第 29 回年次大会 発表論文集, pp. 512–515, 2023.

[2] Annemarie Friedrich, Heike Adel, Federico Tomazic, Johannes Hingerl, Renou Benteau, Anika Marusczyk, and Lukas Lange. The SOFC-exp corpus and neural approaches to information extraction in the materials science domain. In **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics**, pp. 1255–1268. ACL, 2020.

[3] Shu Huang and Jacqueline M. Cole. A database of battery materials auto-generated using chemdataextractor. **Scientific Data**, Vol. 7, No. 1, pp. 2052–4463, 2020.

[4] Fabrizio Gilardi, Meysam Alizadeh, and Ma ë l Kubli. Chatgpt outperforms crowd workers for text-annotation tasks. **Proceedings of the National Academy of Sciences**, Vol. 120, No. 30, p. e2305016120, 2023.

[5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In **Advances in Neural Information Processing Systems**, Vol. 33, pp. 1877–1901. Curran Associates, Inc., 2020.

[6] Md Tahmid Rahman Laskar, M Saiful Bari, Mizanur Rahman, Md Amran Hossen Bhuiyan, Shafiq Joty, and Jimmy Huang. A systematic study and comprehensive evaluation of ChatGPT on benchmark datasets. In **Findings of the Association for Computational Linguistics: ACL 2023**, pp. 431–469. ACL, July 2023.

[7] Bart lomiej Koptyra, Anh Ngo, Lukasz Radliński, and Jan Kocoń. Clarin-emo: Training emotion recognition models using human annotation and chatgpt. In **International Conference on Computational Science**, pp. 365–379. Springer, 2023.

[8] Taiki Watanabe, Akihiro Tamura, Takashi Ninomiya, Takuya Makino, and Tomoya Iwakura. Multi-task learning for chemical named entity recognition with chemical compound paraphrasing. 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. 6244–6249. ACL, 2019.

[9] Amalie Trewartha, Nicholas Walker, Haoyan Huo, Sanghoon Lee, Kevin Cruse, John Dagdelen, Alexander Dunn, Kristin A. Persson, Gerbrand Ceder, and Anubhav Jain. Quantifying the advantage of domain-specific pretraining on named entity recognition tasks in materials science. **Patterns**, Vol. 3, No. 4, p. 100488, 2022.

[10] Gupta Tanishq, Zaki Mohd, and N. M. Krishnan. Matscibert: A materials domain language model for text mining and information extraction. **Npj Computational Materials**, Vol. 8, No. 1, pp. 1–11, 2022.

[11] Yu Song, Santiago Miret, and Bang Liu. MatSci-NLP: Evaluating scientific language models on materials science language tasks using text-to-schema modeling. In **Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 3621–3639. ACL, July 2023.

[12] Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. Gptner: Named entity recognition via large language models, 2023.

[13] Israt Jahan, Md Tahmid Rahman Laskar, Chun Peng, and Jimmy Huang. Evaluation of ChatGPT on biomedical tasks: A zero-shot comparison with fine-tuned generative transformers. In **The 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks**, pp. 326–336. ACL, July 2023.

[14] Bernal Jimenez Gutierrez, Nikolas McNeal, Clayton Washington, You Chen, Lang Li, Huan Sun, and Yu Su. Thinking about GPT-3 in-context learning for biomedical IE? think again. In **Findings of the Association for Computational Linguistics: EMNLP 2022**, pp. 4497–4512. ACL, December 2022.

[15] Maciej P. Polak and Dane Morgan. Extracting accurate materials data from research papers with conversational language models and prompt engineering, 2023.

[16] Necva Bölücü, Maciej Rybinski, and Stephen Wan. Impact of sample selection on in-context learning for entity extraction from scientific writing. In **Findings of the Association for Computational Linguistics: EMNLP 2023**, pp. 5090–5107. ACL, December 2023.

[17] Shuohang Wang, Yang Liu, Yichong Xu, Chenguang Zhu, and Michael Zeng. Want to reduce labeling cost? GPT-3 can help. In **Findings of the Association for Computational Linguistics: EMNLP 2021**, pp. 4195–4205. ACL, November 2021.

[18] Bosheng Ding, Chengwei Qin, Linlin Liu, Yew Ken Chia, Shafiq Joty, Boyang Li, and Lidong Bing. Is GPT-3 a good data annotator? In **Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 11173–11195. ACL, July 2023.

[19] Sheshera Mysore, Zachary Jensen, Edward Kim, Kevin Huang, Haw-Shiuan Chang, Emma Strubell, Jeffrey Flanigan, Andrew McCallum, and Elsa Olivetti. The materials science procedural text corpus: Annotating materials synthesis procedures with shallow semantic structures. In **Proceedings of the 13th Linguistic Annotation Workshop**, pp. 56–64. ACL, August 2019.

[20] Keqin Peng, Liang Ding, Qihuang Zhong, Li Shen, Xuebo Liu, Min Zhang, Yuanxin Ouyang, and Dacheng Tao. Towards making the most of ChatGPT for machine translation. In **Findings of the Association for Computational Linguistics: EMNLP 2023**, pp. 5622–5633. ACL, December 2023.

A　プロンプトの全体像

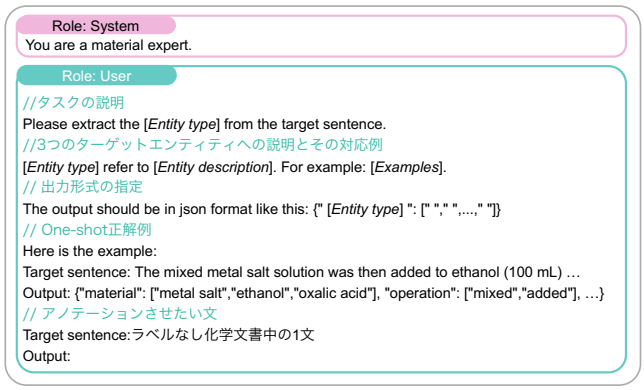

図 2　プロンプトの全体像.

- タスクの説明．実験では，タスクを ”Please extract the [*Entity type*] from the target sentence.” と記述する．[*Entity type*] は抽出したいエンティティの種類を指す．すべてのターゲットエンティティを同時アノテーションする場合，[*Entity type*] には全てのターゲットエンティティを含む．
- 各ターゲットエンティティへの説明とその対応例．エンティティの定義の曖昧さを解消し，GPT により正確な解答を得るために，各エンティティについて詳細な説明を提供する．また，訓練データから各エンティティに対応する例を選択し，説明に加える．
- 出力形式の指定．出力形式を指定するためのプロンプトを設計する．設計意図として，出力形式の維持や出力のラベルへの変換の容易さを確保すること，GPT が生成しやすい形にすることが挙げられる．今回は json 形式を基本となる出力形式にする．これらの基準に基づき，2 つの出力形式を検討する．*a*) トークンとラベルのペアで出力，*b*) json 形式での出力．
- アノテーションを行う文．最後に処理の対象となる文を入力する．

One-shot 正解例については 4.1 節に記載している．ロールについては，本手法で採用した「Material expert（材料分野の専門家）」以外にも，「Annotation machine（アノテーションをする機械）」の場合も検証している．またロールを指定しない「なし」についても比較している．

B　プロンプトの組合せ及び比較

本文に明記できなかったプロンプトの組合せを説明する．精度比較のため 5.1 節で説明した「同時・One-shot 有」 を基盤とする（表中「ベース」）．「AM」と「なし」は，それぞれロールを「Annotation machine」と「なし」とした場合である（ベースは「Material expert」）．ここまでの出力形式はアペンディクス A に示すように，json 形式であった．一方で，本タスクは固有表現抽出であり，このタスクでよく使われる各トークンとラベルのペアを BIO2 タグ形式で出すという方法もある．この出力方法による差を見るために，これを「ペア」とした．

表 3　異なるプロンプトにおけるアノテーション精度 (F1 値).

	ベース	ロール		出力形式
		AM	なし	ペア
MAT	*0.57*	0.56	0.56	0.56
OPE	*0.77*	0.76	0.74	0.70
PRO	*0.14*	0.12	0.14	0.13
Overall	*0.57*	0.56	0.55	0.56

表 3 に各プロンプトにおけるアノテーションの F1 値を示す．まず，ロールの影響について，実験で利用した「Material expert」が最も良い精度であった（表中のベース）．ロールについては「なし」が全体的に最も低い値となった．次に，出力形式の影響を比較する．エンティティの中で一番影響を受けているのは OPE（0.70 vs 0.77）だが，全体的な影響は小さい．ただし，トークン長の面では json 形式の方が短くなる傾向があるため，コストの面で大きなメリットがある．