敵対的マルチエージェントシステムによる階層的テキスト分類

渡辺 建翔 [1]　福本 文代 [2]
[1] 山梨大学工学部　[2] 山梨大学大学院総合研究部工学域
{t22cs056,fukumoto}@yamanashi.ac.jp

概要

階層的テキスト分類は，非階層的な手法と比べ詳細，かつ体系的な情報抽出が可能である反面，下位カテゴリにおけるデータが希薄であるため，ノイズに対する脆弱性が深刻な問題となる．この問題を解決するためには，少量データにおいてもノイズに惑わされず，文脈の本質的な特徴を学習する堅牢性が求められる．本研究は，分類器と攻撃モデルを反復的に競合させる Hierarchical Text Classification with ATM (HTCATM) を提案する．HTCATM は，攻撃モデルが生成する敵対的サンプルを用いた反復学習により，下位カテゴリにおいても高い識別能力と堅牢性を目指す．実験の結果，提案手法はベースラインである BERT [1] の精度を上回り，敵対的チューニングによる有効性が確認できた．

1　はじめに

テキスト分類は，災害時の情報分析やニュース分類など，現代社会における情報処理の重要な基盤技術である．特に，災害発生時の救助要請や被害状況を瞬時かつ正確に把握するためには，単に「災害関連」と分類するだけでなく，具体的な「被害の種類」や「緊急度」といった詳細なカテゴリに細分化し理解する必要がある．この要求に応えるため，テキストをカテゴリをノードとする階層構造へ分類する階層的テキスト分類 (Hierarchical Text Classification; HTC) に関する研究が盛んに行われている [2][3][4]．しかし，階層構造を用いた分類には，階層構造に起因するデータ分布の不均衡，すなわち，階層の上位に位置する一般的なカテゴリに比べ，下位に位置する詳細なカテゴリほど利用可能な訓練データが極端に少なくなる「データ希薄性」の問題が存在する．加えて，多くの HTC タスクは一つのテキストに対し複数のカテゴリを付与するマルチラベル分類として定式化されるため，正解ラベルの組み合わせが複雑化し，データが希薄な下位カテゴリにおける分類はさらに困難となる．現実の SNS や速報記事には多くのノイズや誤情報が含まれており，データが希薄な下位カテゴリでは，これらのノイズにより決定境界が不安定になりやすく，BERT などの従来の分類モデルでは分類精度が著しく低下するという問題が指摘されている [5]．本研究は，このようなデータ希薄な階層構造の下位カテゴリにおいても，ノイズや誤情報に対して堅牢に動作するテキスト分類モデルを構築することを目的とする．我々は Zhu ら [6] により提案された ATM (Adversarial Tuning Multi-agent System) の枠組みに着想を得た．Zhu らは，検索拡張生成 (RAG) において，検索された文書に含まれる誤情報や虚偽内容（fabrications）が LLM のハルシネーションを引き起こす問題に着目し，ATM を提案した [6]．ATM は，回答生成を行う Generator と，それを惑わすための巧妙な虚偽情報を生成する Attacker という 2 つの Agent で構成される．これらを交互に競わせる反復的敵対的チューニング (Iterative Adversarial Tuning) を行うことで，Attacker はより判別困難なノイズを生成するように，Generator はそのノイズを識別し無視するように進化し，結果として検索拡張生成において高い堅牢性を獲得する．本研究では，この ATM の概念を階層的テキスト分類タスクに応用した HTCATM を提案する．具体的には，階層化された全カテゴリを対象とする BERT ベースの分類器 (Classifier) と，分類器を不正解に導くような敵対的サンプルを生成する攻撃モデル (Attacker) を構築し，これら 2 つのモデルを敵対的にチューニングする．特に，階層構造における「兄弟ノード」や「従兄弟ノード」など，意味的に近接し誤分類しやすいカテゴリへの誘導を行うことで，階層特有の難しさを考慮した攻撃を生成する．このプロセスは階層の上下関係に基づく適用順序には依存せず，階層に含まれる全ノードに対して一括して適用される．これにより，分類器は個別の階層レベルにとらわれず，表面的なノイズに左右されない大局的かつ局所的な文脈

の特徴を同時に学習し，データ希薄な下位カテゴリにおいても堅牢に分類できる能力の獲得を目指す．

2 提案手法

2.1 モデル概要

本研究では，Zhu ら [6] の ATM の枠組みを階層的テキスト分類に拡張した HTCATM を提案する．本手法は，テキスト分類を行う Classifier と，分類器を誤分類に導く敵対的サンプルを生成する Attacker の 2 つの Agent から構成される．Classifier には BERT を採用し，Attacker には偽情報生成能力の高い LLM である Mistral-7B を採用する．HTCATM の核心は，これら 2 つのコンポーネントを反復的に競合させるマルチ反復チューニングにある．Classifier は Attacker が生成する攻撃に対する耐性を高め，Attacker は現在の Classifier の弱点を突く，より巧妙なサンプルを生成するように相互に学習を進める．特に，階層分類の特性を考慮し，Attacker は正解カテゴリの「兄弟」や「従兄弟」にあたる近接カテゴリへ誘導するターゲット攻撃を行う．これにより，単なるランダムノイズではない，意味的に識別が困難な敵対的サンプルを生成し，データ希薄領域における決定境界の明確化を図る．

2.2 マルチ反復チューニング

Step 1: Attacker の DPO 最適化

現在の Classifier を攻撃対象 (スコアラー）として固定する．Attacker を用い，入力テキスト x に対する敵対的候補テキストを複数生成する．これらの候補を Classifier に入力し，各サンプルの不確実性 U を算出する．U は Classifier の分類混乱度を示す指標であり，各カテゴリに関する Binary Cross Entropy (BCE) 損失の総和として定義される．

$$U = \sum_{i=1}^{C} BCE(y_i, l_i) \tag{1}$$

ここで，y_i は正解カテゴリ，l_i は Classifier の出力ロジットである．算出された U に基づき，候補群をランク付けする．Classifier を最も混乱させた（U が最大）候補を x_{chosen}，最も混乱させなかった（U が最小）候補を x_{rejected} として選好ペアを構築する．このデータセットを用い，DPO (Direct Preference Optimization) [7] により Attacker をファインチューニングする．DPO の損失関数は以下の通りである．

$$\mathcal{L}_{\text{DPO}}(A_\theta; A_{\text{ref}}) = -\log\sigma\left(\beta\log\frac{A_\theta(x_{\text{chosen}}|x)}{A_{\text{ref}}(x_{\text{chosen}}|x)} - \beta\log\frac{A_\theta(x_{\text{rejected}}|x)}{A_{\text{ref}}(x_{\text{rejected}}|x)}\right) \tag{2}$$

ここで，σ は logistic sigmoid function，β は参照モデル A_{ref} からの逸脱を制御するハイパーパラメータである．A_θ は学習対象である Attacker，A_{ref} は学習前の参照モデルを表し，$A(y|x)$ は入力 x に対しテキスト y を生成する確率を示す．この学習により，Attacker は Classifier の弱点をより効果的に攻撃し，不確実性を最大化させるテキストを生成するよう最適化される．

Step 2: Classifier の堅牢化学習

Classifier の堅牢化を行うため，最適化された Attacker を用いて元の学習データ（clean text）に対応する敵対的サンプルを生成する．Classifier は clean text，及び敵対的サンプルに clean text を結合したサンプルのペアを同時に学習する．本手法では，敵対的攻撃に対する堅牢性の向上に加え，入力のわずかな変化により予測結果が大きく変動しないようにするため，Miyato ら [8] を参考に，clean text と敵対的サンプルの予測分布間の整合性損失 (Consistency Loss) を導入する．すなわち最終的な総損失 $\mathcal{L}_{\text{total}}$ は，clean text に対する分類損失 $\mathcal{L}_{\text{clean}}$，敵対的サンプルに対する分類損失 $\mathcal{L}_{\text{adv}}$，および整合性損失 $\mathcal{L}_{\text{cons}}$ の加重和とし式 (3) で定義される．

$$\mathcal{L}_{\text{total}} = \lambda_{\text{clean}}\mathcal{L}_{\text{clean}} + \lambda_{\text{robust}}\mathcal{L}_{\text{adv}} + \lambda_{\text{cons}}\mathcal{L}_{\text{cons}} \tag{3}$$

ここで $\lambda_{\text{clean}}, \lambda_{\text{robust}}, \lambda_{\text{cons}}$ は各項の寄与度を制御する重み係数である．学習を安定させるため，λ_{robust} には学習の進行に伴い係数を 0 から目標値まで徐々に増加させるウォームアップ方式を適用した．

2.3 マルチラベル分類

推論時には，親カテゴリの予測結果による探索の制約（枝刈り等）を設けず，全カテゴリに対して独立に閾値判定を行う手法を採用した．具体的には，Sigmoid 関数を適用した各カテゴリの予測確率値が，設定された閾値を超えたものを予測結果とした．

表 1 実験で用いたデータ

データ	$\|L\|$	Depth	Avg($\|L_i\|$)	Train	Val	Test
RCV1	103	4	3.24	20,833	2,316	781,274
WOS	141	2	2.0	30,070	7,163	9,397
NYT	166	8	7.6	23,345	5,834	7,292

3 実験

3.1 実験設定

データセット 実験では，階層型テキスト分類のベンチマークである WOS [13], RCV1 [14], および NYT [15] の 3 種類のデータセットを使用した．各データセットを表 1 に示す．データの分割について，NYT と RCV1 については HiAGM [9] で公開されている分割 (train/dev/test) を使用した．一方，WOS については，元のデータセットの分割比率を保持しながらランダムに再分割して使用した．

比較モデルと設定 HTCATM の Classifier には BERT(bert-base-uncased) を，Attacker には偽情報生成能力が高い Mistral-7B-Instruct-v0.2 [16] を使用した．これらのモデルはいずれも Hugging Face Transformers ライブラリ [17] を用いて実装した．比較対象として，本手法の初期チューニングを行った BERT に加え，階層型テキスト分類の最新の手法である HiSR [12] を含む 4 手法を用いた．評価指標には Micro-F1 および Macro-F1 を使用した．特にデータ不均衡の影響を受けやすい下位カテゴリの精度を測るため，本研究では Macro-F1 を重視する．

3.2 実験結果

ベースラインとの比較 表 2 に，各モデルに対し閾値を 0.5 に固定して評価を行った結果を示す．表 2，ベースラインである BERT と比較し，RCV1 および NYT データセットにおいて提案手法 HTCATM が Micro/Macro-F1 の双方で上回る精度を示した．特に NYT では，Macro-F1 において BERT の 60.59 から HTCATM（Iter4）の 64.82 へと，4.23 ポイントの顕著な改善が確認できる．また，RCV1 においても反復学習が進むにつれて精度が向上し，Iter4 時点では 62.77 を記録して BERT の 59.63 を上回った．一方，WOS においては BERT が Macro-F1 で 82.36 と高いスコアを示し，提案手法の最高値（Iter3: 82.03）はわずかに及ばなかった．しかし，WOS においても Iter0 の 80.96 から Iter3 の 82.03 へと，反復回数が増えるにつれて着実に精度が向上しており，敵対的チューニングによるモデルの改善効果は確認できた．一方で，SOTA 手法である HiSR や NERHTC と比較すると，全体として HTCATM のスコアは及ばない結果となった．これは，HiSR や NERHTC がラベル間の親子関係を捉えるための特殊なグラフ構造や外部知識を組み込んでいるのに対し，本手法は BERT をベースとした単純な分類器構造を用いているためであると考えられる．しかし，モデル構造を複雑化することなく，反復学習のみにより BERT の性能を大きく引き上げ，これらのモデルとの差を縮めた点は，Attacker が生成する敵対的サンプルを通じて Classifier の決定境界における脆弱性を露呈させ，それを重点的に修正・強化する本手法の学習プロセスの有効性を示唆している．

閾値最適化による評価 表 3 に，検証データを用いて Macro-F1 が最大となるように閾値を最適化（探索範囲 [0.1, 0.6]）した結果の精度を示す．RCV1 および NYT データセットにおいて，HTCATM はベースラインである BERT を上回る精度を達成した．具体的には，NYT において BERT が 64.52 であるのに対し，HTCATM（Iter4）では 66.99 となり，2.47 ポイントの向上が見られた．RCV1 でも同様に，BERT の 63.51 から HTCATM（Iter4）の 65.09 へと精度が向上している．一方で，SOTA である HiSR と比較すると，HiSR は全てのデータセットにおいて最高精度（RCV1: 69.70, WOS: 83.28, NYT: 69.28）を記録しており，提案手法はこれには及ばなかった．特に WOS においては，HiSR が 83.28，BERT が 82.65 であるのに対し，提案手法は Iter3 で 82.01 にとどまり，両者をわずかに下回る結果となった．しかし，HiSR のような特化型のモデル構造を持たないにもかかわらず，特に階層構造が深く，かつ 1 文書あたりのカテゴリ数が多い RCV1 や NYT といった難易度の高いデータセットにおいて BERT を上回った事実は重要である．これは，敵対的チューニングによってモデルが決定境界付近の難しいサンプル（データ希薄な下位カテゴリなど）をより正確に識別できるようになったためと考えられる．

階層レベル別の分類精度 表 4 に，各データセットの最上位層（L1）および最下層（Deepest）における Macro-F1 スコアの推移を示す．階層構造が比較的浅い WOS（L2）や RCV1（L4）においては，階層構造を明示的にモデル化している HiSR が依然として優位であった．しかし，階層が深く最も難易度の高い NYT（L8）においては，提案手法である

表 2　提案手法とベースラインとの精度比較: 表中太字は最高精度を示す.

モデル	RCV1		WOS		NYT	
	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1
BERT	86.01	59.63	**88.09**	**82.36**	79.49	60.59
HiAGM [9]	83.96	63.35	85.82	80.28	74.97	60.83
HALB [10]	86.94	69.32	87.45	82.04	79.56	69.28
NERHTC [11]	87.50	69.76	87.42	81.93	**80.97**	**70.99**
HiSR [12]	**87.59**	**70.72**	87.52	82.04	80.32	70.11
HTCATM (Iter0)	86.41	58.96	87.07	80.96	79.52	62.81
HTCATM (Iter1)	86.70	60.29	87.30	81.45	79.80	64.54
HTCATM (Iter2)	86.70	61.15	87.44	81.79	79.83	64.58
HTCATM (Iter3)	86.70	60.29	87.50	82.03	79.63	64.31
HTCATM (Iter4)	86.77	62.77	87.10	81.68	79.88	64.82

表 3　閾値最適化による精度比較

モデル	RCV1	WOS	NYT
	Macro-F1	Macro-F1	Macro-F1
BERT	63.51	82.65	64.52
HiSR[Zhou’25]	**69.70**	**83.28**	**69.28**
HTCATM (Iter0)	62.86	81.22	65.68
HTCATM (Iter1)	63.56	81.45	66.44
HTCATM (Iter2)	63.93	81.79	66.76
HTCATM (Iter3)	64.76	82.01	66.72
HTCATM (Iter4)	65.09	81.77	66.99

表 4　最上位層 (L1) と最下層 (Deepest) における Macro-F1

モデル	WOS (L2)		RCV1 (L4)		NYT (L8)	
	L1	Deepest	L1	Deepest	L1	Deepest
HiSR	**92.48**	**82.80**	**93.00**	**83.25**	89.04	54.46
HTCATM(iter0)	91.81	80.66	92.61	79.83	89.52	50.31
HTCATM(iter1)	91.96	80.90	92.81	82.18	**89.66**	57.73
HTCATM(iter2)	92.13	81.25	92.78	82.06	89.42	**59.06**
HTCATM(iter3)	91.96	81.49	92.65	81.92	89.54	57.28
HTCATM(iter4)	91.83	81.24	92.80	81.40	89.59	56.01

HTCATM が HiSR を凌駕する結果を示した．特に NYT の最下層（Deepest）に着目すると，ベースラインとなる Iter0 の時点では 50.31 と HiSR の 54.46 を下回っていたが，Iter1 以降で逆転し，Iter2 では 59.06 という最高値を記録した．Iter4 においても 56.01 と，依然として SOTA を上回る精度を維持している． 一般に最下層のノードは学習データが希薄であり分類が困難であるが，NYT のような深く複雑な階層構造において Iter を重ねるごとに精度が大幅に向上していることから，敵対的学習の導入が決定境界の堅牢性を高め，特にデータ希薄な下位カテゴリの識別能力向上に強く寄与していると言える．

4　おわりに

本研究は，階層的テキスト分類におけるデータ希薄性およびノイズへの堅牢性を向上させるため，RAG の堅牢化手法である ATM に着想を得た敵対的チューニング手法 HTCATM を提案した．BERT をベースとした Classifier と，LLM を用いた Attacker を反復的に競合させることで，モデルがノイズに惑わされず，本質的な特徴を学習できる枠組みを構築した．3 つのベンチマークデータセットを用いた評価実験により，以下の結論が得られた．第一に，HTCATM はベースラインである BERT と比較し，RCV1 および NYT において一貫して高い分類精度を達成した．特に NYT では，閾値最適化後の Macro-F1 で 2.47 ポイントの向上を確認した．第二に，SOTA モデルである HiSR との比較により，敵対的チューニングの有効性が示された．WOS のような浅い階層構造では HiSR が優位であったが，階層が深い NYT の最下層では，HTCATM がより高い精度を示した．特に，反復学習に伴い最下層の精度が大幅に向上（Iter0 の 50.31 から Iter2 の 59.06 へ）した結果は，HTCATM の有効性を示している．以上の結果から，HTCATM は，階層構造に特化したアーキテクチャを用いずとも敵対的学習のみを用い BERT の能力を引き出し，下位カテゴリの識別において優れた精度を発揮することが示された．今後の課題としては，SOTA である HiSR の精度を全域にわたって凌駕するため，2 点を検討する．先ず，HiSR のような階層構造を明示的に取り込むモデルへの本手法の統合である．これにより，浅い階層から深い階層まで全域にわたって SOTA を上回る，より堅牢で高精度なモデルの構築が期待される．2 点目は，データセットの特性に応じた攻撃戦略の最適化である．WOS のように兄弟ノード間の意味的距離が遠い場合，現在のルールベースなターゲット攻撃では有効でない可能性がある．そこで，埋め込み空間における意味的な類似度に基づきターゲットを動的に決定するなど Attacker の戦略を高度化することにより，階層構造の深さに依存しない堅牢化を目指す予定である．

謝辞

本研究は, 科研費 24K15085, 及びテレコム先端技術研究支援センターの助成を受けたものです.

参考文献

[1] 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 (NAACL-HLT)**, 2019.

[2] Fabian Karl and Ansgar Scherp. Hydra: A multi-head encoder-only architecture for hierarchical text classification. In **Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing**, 2025.

[3] Mingxuan Xia, Zhijie Jiang, Haobo Wang, Junbo Zhao, Tianlei Hu, and Gang Chen. Ensembling prompting strategies for zero-shot hierarchical text classification with large language models. In **Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing**, 2025.

[4] Qian Zhang, Qinliang Su, Wei Zhu, and Yachun Pang. Hierprompt: Zero-shot hierarchical text classification with llm-enhanced prototypes. In **Findings of the Association for Computational Linguistics: EMNLP 2025**, 2025.

[5] Guilherme Fonseca, Gabriel Prenassi, Washington Cunha, Marcos André Gonçalves, and Leonardo Rocha Rocha. Instance-selection-inspired undersampling strategies for bias reduction in small and large language models for binary text classification. In **Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, 2025.

[6] Junda Zhu, Lingyong Yan, Haibo Shi, Dawei Yin, , and Lei Sha. Atm: Adversarial tuning multi-agent system makes a robust retrieval-augmented generator. In **Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, 2024.

[7] Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In **Advances in Neural Information Processing Systems (NeurIPS)**, 2023.

[8] Takeru Miyato, Shin ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: A regularization method for supervised and semi-supervised learning. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, 2018.

[9] Jie Zhou, Chunping Ma, Dingkun Long, Guangwei Xu, Ning Ding, Haoyu Zhang, Pengjun Xie, and Gongshen Liu. Hierarchy-aware global model for hierarchical text classification. In **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics**, 2020.

[10] Jun Zhang, Yubin Li, Fanfan Shen, Chenxi Xia, Hai Tan, and Yanxiang He. Hierarchy-aware and label-balanced model for hierarchical text classification. **Knowledge-Based Systems**, 2024.

[11] Fuhan Cai, Duo Liu, Zhongqiang Zhang, Ge Liu, Xiaozhe Yang, and Xiangzhong Fang. Ner-guided comprehensive hierarchy-aware prompt tuning for hierarchical text classification. In **Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)**, 2024.

[12] Juncheng Zhou, Lijuan Zhang, Yachen He, Rongli Fan, Lei Zhang, and Jian Wan. A novel negative sample generation method for contrastive learning in hierarchical text classification. In **Proceedings of the 31st International Conference on Computational Linguistics (COLING 2025)**, 2025.

[13] Kamran Kowsari, Donald E. Brown, Mojtaba Heidarysafa, Kiana Jafari Meimandi, Matthew S. Gerber, and Laura E. Barnes. Hdltex: Hierarchical deep learning for text classification. In **2017 IEEE International Conference on Machine Learning and Applications**, 2017.

[14] David D. Lewis, Yiming Yang, Tony G. Rose, and Fan Li. Rcv1: A new benchmark collection for text categorization research. **Journal of Machine Learning Research**, 2004.

[15] Evan Sandhaus. The new york times annotated corpus. **Linguistic Data Consortium**, 2008.

[16] Albert Q. Jiang, Alexandre Sablayrolles, et al. Mistral 7b. **arXiv preprint arXiv:2310.06825**, 2023.

[17] Thomas Wolf, et al. Transformers: State-of-the-art natural language processing. In **Proceedings of EMNLP 2020: System Demonstrations**, 2020.