埋め込みベクトルへの効果的な変更に基づく頑健な学習手法の実現

理学専攻・情報科学コース　田屋 侑希　(指導教員：小林 一郎)

1　はじめに

近年，自然言語処理の分野において，汎用言語モデル（BERT，RoBERTa 等）は様々なタスクで大きな成功を収めている．一方で，入力データの変化に敏感であることが知られている．入力データの変化に対するモデルの頑健性を向上させるため，敵対的学習という学習手法が提案された [1]．本研究では，摂動の代わりに類似している単語の方向にノイズを加えることで，予測性能が敵対的学習と同等のモデルを提案する．また，ノイズの加える割合や類似している単語の選び方の違いによる効果を検証する．

2　敵対的学習

本研究では提案手法の比較対象として SMART [2] を利用した．SMART は仮想敵対的学習 [3] を用いており，目的関数は式 1 のとおりである．

$$\min_{\theta} \mathbb{E}_{(x,y)\sim D}[l(f(x;\theta),y)+ \alpha \max_{\delta} l(f(x+\delta;\theta),f(x;\theta))] \tag{1}$$

この式は，データセット D の入力 x に初期摂動 δ を加算してモデル f に入力し，自身のモデルの予測結果 $f(x;\theta)$ との誤差が最大になる摂動 δ を求め，再び $x+\delta$ をモデルに入力し，モデルの全体の損失 $\mathbb{E}$ が最小になるパラメータ θ を求めることを意味する．この式における，l を敵対的損失と呼び，制約付き勾配降下法（PGD）[1] を解くことで δ を求めている．ここでの制約は摂動の大きさである．仮想敵対的学習における敵対的損失 l は，入力データ x とその近傍のデータ $x+\delta$ はモデルが同じ予測をするという制約を意味する正則化項とみなすことができる．また，式 1 の α は一般的な損失と敵対的損失のトレードオフを制御するハイパーパラメータで，本研究では SMART と同様，1.0 に設定した．

3　提案手法

図 1 における Step2 の Token Embeddings として効果的な単語埋め込みを検証するため，提案手法を**実験 1** から**実験 3** に分けて説明する．

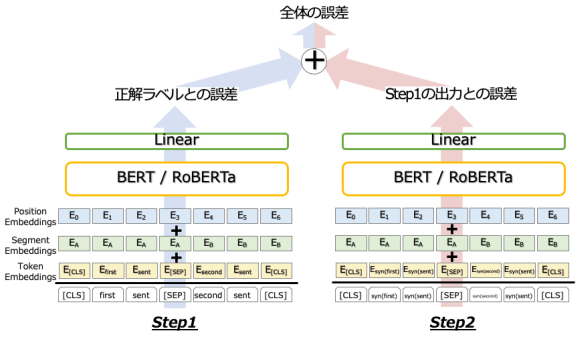

図 1: 提案手法の概要図

3.1　実験 1：類似単語の方向にノイズ付加

Step1 で入力したそれぞれの Token Embeddings（$\boldsymbol{E}_{\text{original}}$）に対して，最もコサイン類似度が高い単語の Token Embeddings（$\boldsymbol{E}_{\text{similar}}$）を求めて，これらの差ベクトル（$\boldsymbol{E}_{\text{diff}}$）を計算する．

$$\boldsymbol{E}_{\text{diff}} = \boldsymbol{E}_{\text{similar}} - \boldsymbol{E}_{\text{original}}$$

Step1 の Token Embeddings に差ベクトル（$\boldsymbol{E}_{\text{diff}}$）を足し，新しい Token Embeddings（$\boldsymbol{E}_{\text{new}}$）を作成する．これを Step2 の入力とする．

$$\boldsymbol{E}_{\text{new}} = \boldsymbol{E}_{\text{original}} + noise\_size \times \boldsymbol{E}_{\text{diff}}$$

ここで，類似度が高い単語の Token Embeddings にどの程度近づけるかを，$noise\_size$ 係数（$0 \leq noise\_size \leq 1$）を用いて決定する．$noise\_size$ が 1 の場合は，コサイン類似度が高い単語の Token Embeddings を Step2 に入力していることを意味する．本手法では，$noise\_size$ を 0.5 に設定した（**提案手法 1-1**）．また，本研究では，誤差逆伝播を用いて求める**摂動**と区別するため，Step1 の Token Embeddings に加算される $noise\_size \times \boldsymbol{E}_{\text{diff}}$ を**ノイズ**と定義する．提案手法 1-1 では，常時コサイン類似度が一番高い単語が選ばれてしまうため，コサイン類似度が高い上位 10 単語の中からランダムに選択して $\boldsymbol{E}_{\text{similar}}$ とする設定でも実験を行った（**提案手法 1-2**）．

3.2　実験 2：部分的な単語にノイズ付加

次に全部の単語ではなく一部の単語にノイズを加える実験を行う．ノイズを加える単語を部分的に（1 文の 15% の単語にノイズ付加）することの効果を確認する．ノイズの加え方は提案手法 1-1 と同様の手法を用いる．新しい Token Embeddings（$\boldsymbol{E}_{\text{new}}$）に置き換える単語の決め方は，下記の 3 通りを実験した．

提案手法 2-1　ランダム
提案手法 2-2　Saliency の値が高い単語
提案手法 2-3　Saliency の値が低い単語

提案手法 2-2，2-3 における Saliency の値とは，Step1 の誤差を用いて，Step1 の Token Embeddings に対する勾配を計算し，勾配の絶対値を埋め込みベクトルの次元数分足し合わせたものである．一般的に，Saliency の値は予測結果の判断根拠を解釈するために利用され，Saliency の値が高いほど，予測結果に影響を与えている単語であると解釈することができる．

3.3　実験 3：WordNet を用いたノイズ付加

次に，Step2 に入力する単語として，Step1 の類義語またはコサイン類似度が高い単語のどちらが効果的かを確認する．図 1 の Step1 で入力した単語の類義語を WordNet で検索し，類義語が存在する場合は類義語を Step2 に入力する（**WordNet_1.0**）．類義語が複数存在する場合は，ランダムに 1 つ選んで入力する．この時，Step2 の Token Embeddings は Step1 の単語の類義語そのものを意味するため，実験 1 における $noise\_size$ は 1.0 と解釈できる．また，提案手法 1-1 で $noise\_size$ を 1.0 とした場合，つまり Step2 でコサイン類似度が最も高い単語を入力した場合の実験（**提案手法 1-1_1.0**）も行い比較する．さらに，WordNet を用いて，Step2 に入力する類義語を決定し，$noise\_size$ を 0.5 に設定した実験も行った（**WordNet_0.5**）．

4 実験

4.1 実験設定

データセットは ANLI（Adversarial Natural Language Inference）[4] を利用した．ANLI は A1，A2，A3 の 3 つのデータセットからなり，汎用言語モデル（BERT，RoBERTa）が予測を誤るデータを積極的に検証データとテストデータに集めたものである．評価指標は正解率を用いる．また，SMART の実験方法に倣い，ANLI（A1・A2・A3）の訓練データを用いて学習し，A1・A2・A3 それぞれの検証データ，テストデータに対して評価を行った．実装は，MT-DNN をもとに行い，事前学習済み言語モデルは $\text{BERT}_{\text{BASE}}$(uncased)，及び $\text{RoBERTa}_{\text{LARGE}}$ を用いた．ファインチューニング の設定として，学習率は 2×10^{-5}，バッチサイズは 32（$\text{BERT}_{\text{BASE}}$），16（$\text{RoBERTa}_{\text{LARGE}}$），エポック数は 6，最適化手法は Adam を用いた．先行研究の SMART [2] における摂動の大きさ等はデフォルトの設定をそのまま利用した．

4.2 実験結果・考察

A1・A2・A3 それぞれのテストデータで評価し，その算術平均を取った値を表 1，表 2 に記す．また，実験結果は異なるシード値で 6 回実験を行い，その評価結果の平均を取った値である．表 1，表 2 の 1，2 行目は通常のファインチューニングと SMART の再実験結果である．以降，テストデータにおける正解率について言及する．

表 1: 実験結果（実験 1・実験 2 における正解率）

	$\text{BERT}_{\text{BASE}}$		$\text{RoBERTa}_{\text{LARGE}}$	
	検証	テスト	検証	テスト
fine-tuning	48.9	48.8	57.5	56.2
+SMART	49.0	49.2	58.2	56.5
+提案手法 1-1	48.8	49.5	57.7	56.2
+提案手法 1-2	48.8	**49.8**	57.0	56.3
+提案手法 2-1	48.7	49.2	57.6	56.6
+提案手法 2-2	48.4	49.5	57.3	56.2
+提案手法 2-3	48.9	48.7	58.1	**56.7**

実験 1・実験 2　表 1 より，$\text{BERT}_{\text{BASE}}$ において提案手法 1-1，1-2 は，通常のファインチューニング，SMART と比較して正解率が向上した．提案手法 2-1，2-2，2-3 の結果からは，部分的にノイズを加えるよりも文全体にノイズを加える方（提案手法 1-1）が効果的であることが確認できた．また，Saliency の値が低い単語より高い単語にノイズを加える方が効果があった．$\text{RoBERTa}_{\text{LARGE}}$ では，提案手法 1-1，1-2 は通常のファインチューニングと同等の結果となった．一方，部分的にノイズを加えた提案手法 2-1，2-3 は SMART より正解率が向上した．また，難易度が一番高い A3 のデータセットに着目すると，$\text{BERT}_{\text{BASE}}$ と $\text{RoBERTa}_{\text{LARGE}}$ のどちらにおいても，Saliency の値が高い単語にノイズを加えた場合に精度が高くなることが確認できた．

表 2: 実験結果（実験 3 における正解率）

	$\text{BERT}_{\text{BASE}}$		$\text{RoBERTa}_{\text{LARGE}}$	
	検証	テスト	検証	テスト
fine-tuning	48.9	48.8	57.5	56.2
+SMART	49.0	49.2	58.2	56.5
+WordNet_1.0	47.7	48.7	57.0	56.1
+提案手法 1-1_1.0	48.0	49.0	56.7	55.2
+WordNet_0.5	49.0	48.9	58.1	57.1

実験 3　表 2 から，$\text{BERT}_{\text{BASE}}$ において Step2 に入力する単語は，WordNet を用いて類義語に置き換える（WordNet_1.0）より Step1 の単語埋め込みとのコサイン類似度が高い単語に置き換えた方（提案手法 1-1_1.0）がやや効果があることが確認できた．また，提案手法 1-1 と同様，WordNet を用いた場合においても，類義語そのもの（WordNet_1.0）より類義語との中間に位置するように Step2 の Token Embeddings を作成した方（WordNet_0.5）が正解率が高いことが確認できた．$\text{RoBERTa}_{\text{LARGE}}$ においては，$\text{BERT}_{\text{BASE}}$ の結果とは異なり，コサイン類似度が高い単語よりも WordNet における類義語の方が正解率が高い結果となった．WordNet を用いて類義語に置き換える際，類義語がない単語も存在するため，結果的に 1 文の約 40%（$\text{BERT}_{\text{BASE}}$ では約 30%）の単語が置換されている．実験 2 の結果も踏まえて，$\text{RoBERTa}_{\text{LARGE}}$ では部分的にノイズを加えることに効果があると考えられる．*noise_size* に関しては，$\text{BERT}_{\text{BASE}}$ と同様，0.5 にした方（WordNet_0.5）が類義語そのもの（WordNet_1.0）より正解率が高くなることが確認でき，SMART よりも大きく精度が向上した．

ノイズの大きさ　単語埋め込みに対する摂動またはノイズの大きさ（L2 ノルム）の割合を計算した．SMART における摂動は約 0.003%，提案手法 1-1 のノイズは約 30%，WordNet_0.5 のノイズは約 70%であり，ノイズの大きさを大きくしてもノイズの方向性に意味がある場合，精度の低下がおこらないことが確認できた．

5 おわりに

本研究では，敵対的学習における摂動の代わりに，類似している単語の方向にノイズを加えることで仮想敵対的学習である SMART と同等の精度が出ることを確認した．また，通常のファインチューニングと比較して順伝播が 2 回，逆伝播が 1 回多い敵対的学習を，順伝播が 1 回のみ多いモデルに置き換えることができ，計算時間を約 23%削減することができた．敵対的学習において，摂動ではなく図 1 のように誤差を 2 つ足し合わせる学習手法が精度向上に貢献していることを示唆する 1 つの手法を提案した．一方，提案手法のノイズの大きさが，モデルの予測性能向上に有効である証拠が不十分であるため，提案手法のノイズの汎用性について検証していきたい．また，$\text{BERT}_{\text{BASE}}$ と $\text{RoBERTa}_{\text{LARGE}}$ で異なる傾向の実験結果が得られたため，各モデルの特徴や埋め込み空間の調査を継続したい．

参考文献

[1] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

[2] Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Tuo Zhao. Smart: Robust and efficient fine-tuning for pre-trained natural language models through principled regularized optimization. *arXiv preprint arXiv:1911.03437*, 2019.

[3] 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*, Vol. 41, No. 8, pp. 1979–1993, 2018.

[4] Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4885–4901, Online, July 2020. Association for Computational Linguistics.