教師あり文章埋め込みに対する敵対的正則化の効果

福地成彦　　田中久美子
東京大学
情報理工学系研究科　先端科学技術研究センター
fukuchi@cl.rcast.u-tokyo.ac.jp,kumiko@i.u-tokyo.ac.jp

1　はじめに

単語埋め込み (word embedding)[9] に代表される自然言語のベクトル表現は、自然言語処理の多くの分野で広く活用されている。その中でも、近年、文章埋め込み (sentence embedding) は、文章をベクトルに変換する手法として盛んに研究が行われている。文章埋め込み手法には、教師ありと教師なしの二つアプローチがあり、前者の代表的な手法として自然言語推論 (Natural Language Inference, NLI) を用いた InferSent[3] がある。本研究では、敵対的正則化 (adversarial regularization) が言語の表現学習に有効かを確かめるために、InferSent[3] のに対して自然言語推論の学習に敵対的学習 (adversarial training)[5] と virtual adversarial training(VAT[8] を行い、本稿では、その効果を調べた得られた文章埋め込みを SentEval[4, 2] で評価した結果を報告する。

従来の言語処理における敵対的正則化の研究の多くでは、end-to-end のタスクでの学習が議論されてきた。これに対し本研究では、タスクの中の重要な部分の一つとなる、文章埋め込み表現に対する敵対的正則化の効果を議論する。敵対正則化によって性能の良い文章埋め込み表現を獲得することができれば、それは様々な別のタスクに応用することができる。

2　文章埋め込み

文章埋め込み (Sentence embedding) は x_i を 1 単語とし、文章列 $X = x_1, x_2 \dots x_n$ から d 次元のベクトル $\boldsymbol{v} \in \mathbb{R}^d$ を得る操作として定式化がされる。近年提案されている文章埋め込み手法では、$\boldsymbol{v}$ を得る手段として LSTM などのニューラルネットワークを用いたエンコーダーを用いる場合が多い [4]。これらの手法では、まず、各単語列 $x_1, \dots x_n$ に対応する学習済みの単語ベクトル $\boldsymbol{e}_1, \dots \boldsymbol{e}_n$ を計算する。その後、ニューラルネットワークを用いたエンコーダーで文章ベクトル $\boldsymbol{v}$ を得る。エンコーダーの学習には、教師ありと教師なし手法が存在する。その中でも、教師あり単語埋め込みの手法である InferSent[3] は自然言語推論のデータセットである Stanford Natural Language Inference dataset[1] をエンコーダーの学習に用いている。このデータセットは、57 万ペアの英語の文章それぞれに対し、"entailment", "contradiction", "natural"のいずれかのラベルが付与されている。。InferSent は、文章のペア $(X_1, X_2),\ X_1 = x_1^1, \dots x_n^1,\ X_2 = x_1^2 \dots x_n^2$ を同一のエンコーダーに入力し、得られる二つの文章ベクトル $\boldsymbol{v}^1, \boldsymbol{v}^2$ を入力とする。出力として分類器が文章関係のラベルを分類するように学習を行う。なお、[3] のエンコーダーと分類器のネットワークのアーキテクチャを利用し、以下のようにラベル予測 y を計算する。

$$\boldsymbol{v}_1 = \quad \text{Pooling}\left(\text{BiLSTM}\left(\boldsymbol{e}_1^1, \dots \boldsymbol{e}_n^1\right)\right), \quad (1)$$

$$\boldsymbol{v}_2 = \quad \text{Pooling}\left(\text{BiLSTM}\left(\boldsymbol{e}_1^2, \dots \boldsymbol{e}_n^2\right)\right), \quad (2)$$

$$\hat{y} = \text{CLS}\left(\boldsymbol{v}^1, \boldsymbol{v}^2, \|\boldsymbol{v}^1 - \boldsymbol{v}^2\|, \boldsymbol{v}^1 * \boldsymbol{v}^2\right), \quad (3)$$

ただし、CLS は分類器としての全結合ネットワーク、BiLSTM は双方向の LSTM(long-short-term memory) で、$*$ はアダマール積とする。今、N 組みの文章ペアとラベルから構成されるデータセット $\mathcal{D} = \{X_{1,j}, X_{2,j}, Y_j\}_{j=1}^N$ に対して、エンコーダーと分類器のパラメータを θ, ϕ とする。このとき、エンコーダーとモデルの学習は以下のようになる。

$$\theta, \phi = \arg\min_{\theta,\phi} \left(\mathcal{J}(\mathcal{D}, \theta, \phi)\right), \quad (4)$$

$$\mathcal{J}(\mathcal{D}, \theta, \phi) := \frac{1}{|\mathcal{D}|} \sum_{(X_1, X_2, Y) \in \mathcal{D}} l\left(X_1, X_2, Y, \theta, \phi\right), \quad (5)$$

$$l\left(X_1, X_2, Y, \theta, \phi\right) := -\log\left(p\left(Y | X_1, X_2, \theta, \phi\right)\right), \quad (6)$$

なお、$l\left(X_1, X_2, Y, \theta, \phi\right)$ は負の対数尤度とする。

3 敵対的正則化

3.1 敵対的学習

敵対的サンプル (adversarial examples)[5] とは予測モデルの予測誤差が大きくなるような摂動を加えた入力データである。[5] では、この敵対的サンプルを加えた入力を一種のデータ拡張とみなして、敵対的サンプルを用いた学習手法 (Adversarial Training, AdvT) を提案している。敵対的学習を用いた自然言語処理の既存研究として文書分類 [7] や機械翻訳 [10] などがある。自然言語処理では、入力となる単語が離散的であるため、各単語ベクトル $\boldsymbol{e}_i$ に対して摂動 $\boldsymbol{r}$ を加える。

θ をモデルパラメーター、x を入力、y を予測するためのターゲット、l を損失関数とする。文章 1 と文章 2 に対する敵対的摂動 $\hat{\boldsymbol{r}}_1, \hat{\boldsymbol{r}}_2$ は 6 を用いて以下のように生成される。

$$\hat{\boldsymbol{r}}_1, \hat{\boldsymbol{r}}_2 = \underset{\boldsymbol{r}_1, \boldsymbol{r}_2 < \epsilon}{\arg\max}\, l\left(X_1, \boldsymbol{r}_1, X_2, \boldsymbol{r}_2, Y, \theta, \phi\right) \tag{7}$$

ただし、$l\left(X_1, \boldsymbol{r}_1, X_2, \boldsymbol{r}_2, Y, \theta, \phi\right)$ は X_1, X_2 の単語ベクトルに摂動を加えた場合の負の対数尤度、ϵ は摂動の大きさを表す。敵対的 $\hat{\boldsymbol{r}}_1, \hat{\boldsymbol{r}}_2$ の生成は勾配法を用いて以下のように実行する。

$$\hat{\boldsymbol{r}}_1 = \epsilon \frac{\boldsymbol{a}}{\|\boldsymbol{a}\|_2}, \quad \boldsymbol{a} = \nabla_{\boldsymbol{r}_1} l\left(X_1, \boldsymbol{r}_1, X_2, \boldsymbol{r}_2, Y, \theta, \phi\right) \tag{8}$$

$$\hat{\boldsymbol{r}}_2 = \epsilon \frac{\boldsymbol{a}}{\|\boldsymbol{a}\|_2}, \quad \boldsymbol{a} = \nabla_{\boldsymbol{r}_2} l\left(X_1, \boldsymbol{r}_1, X_2, \boldsymbol{r}_2, Y, \theta, \phi\right) \tag{9}$$

得られた摂動に $\hat{\boldsymbol{r}}_1, \hat{\boldsymbol{r}}_2$ に対する損失関数は

$$\mathcal{A}(\mathcal{D}, \theta, \phi) := \frac{1}{|\mathcal{D}|} \sum_{(X_1, X_2, Y) \in \mathcal{D}} l\left(X_1, \hat{\boldsymbol{r}}_1, X_2, \hat{\boldsymbol{r}}_2, Y, \theta, \phi\right) \tag{10}$$

となる。Adversarial training を用い 6 を拡張し、二つの損失関数の和に対してパラメータを学習する。

$$\theta, \phi = \underset{\theta, \phi}{\arg\min}\left(\mathcal{J}(\mathcal{D}\theta, \phi) + \lambda \mathcal{A}(\mathcal{D}, \theta, \phi)\right) \tag{11}$$

λ は二つの損失関数の割合を決めるための、ハイパーパラメーターである。

3.2 Virtual Adversarial Training

Virtual adversarial training (VAT) は敵対的学習と類似した正則化の手法である。敵対的学習がラベルを使って正則化を行うのに対して、VAT は半教師あり手法である。VAT では、対数尤度について敵対的サンプルを生成するのではなく、ネットワークの出力の分布間距離に対して敵対的サンプルを生成する。

$$l_{\mathrm{VAT}}\left(X_1, \boldsymbol{r}_1, X_2, \boldsymbol{r}_2, Y, \theta, \phi\right) = \mathrm{KL}\left(p\left(\cdot | X_1, X_2, \theta, \phi\right) \| \left(p\left(\cdot | X_1, \boldsymbol{r}_1, X_2, \boldsymbol{r}_2, \theta, \phi\right)\right)\right) \tag{12}$$

$\mathrm{KL}(\cdot\|\cdot)$ はカルバック・ライブラー情報量、$p\left(\cdot | X_1, X_2, \theta, \phi\right)$ はモデルの予測分布とする。virtual adversarial training では (6) のではなく、(12) を用いて、(7) の敵対的サンプルを生成する。

$$\hat{\boldsymbol{r}}_{\mathrm{VAT},1}, \hat{\boldsymbol{r}}_{\mathrm{VAT},2} = \underset{\boldsymbol{r}_1, \boldsymbol{r}_2 < \epsilon}{\arg\max}\, l_{\mathrm{VAT}}\left(X_1, \boldsymbol{r}_1, X_2, \boldsymbol{r}_2, \theta, \phi\right) \tag{13}$$

なお、敵対的摂動は分布間距離に [8] の近似を行った上で (8,9) と同様に生成した。VAT によって生成された摂動に対数する損失関数は、

$$\mathcal{R}(\mathcal{D}, \theta, \phi) \tag{14}$$

$$:= \frac{1}{|\mathcal{D}|} \sum_{(X_1, X_2, Y) \in \mathcal{D}} \mathrm{VAT}\left(X_1, \hat{\boldsymbol{r}}_{\mathrm{VAT},1}, X_2, \hat{\boldsymbol{r}}_{\mathrm{VAT},1}\theta, \phi\right) \tag{15}$$

であり、VAT の最終的な損失関数に対する学習は、

$$\theta, \phi = \underset{\theta, \phi}{\arg\min}\left(\mathcal{J}(\mathcal{D}\theta, \phi) + \lambda \mathcal{R}(\mathcal{D}, \theta, \phi)\right) \tag{16}$$

となる。

4 実験

この実験では、InferSent[3] に基づいて敵対的学習, virtual adversarial training を行い、得られた文章埋め込みについて SentEval[4, 2] を用いて評価を行う。

4.1 学習条件

InferSent の学習条件は [3] の実験条件を参考に決定した。学習するデータセットは Stanford Natural Language Inference dataset[1] で、57 万の文章ペアとその関係性を示すラベルから構成される。

自然言語推論を行うニューラルネットワークは、文章埋め込みを行う 1 層双方向 LSTM と文章ベクトルをもとにラベルを分類する全結合ネットワークからなる。文章埋め込み次元 d は 2048 次元とし、双方向 LSTM

の次元も 2048 次元である。(1, 2) の pooling には max pooling を用いる。全結合ネットワークには 512 次元の 3 層の隠れ層を用いる。単語埋め込みには、Common Crawl 840B で学習をした 300 次元の Glove ベクトル [9] を用いた。adersarial traing と VAT のパラメータは $\epsilon = 0.25, 0.5, \quad \lambda = 1$ とした。

学習の最適化には SGD(確率的勾配法) を用い、学習開始時のラーニングレートは 0.1 に、weight decay は 0.99 に設定した。また、各 epoch で検証用データでの精度の減少した場合、ラーニングレートを 5 分の 1 に変更した。バッチサイズは 64、イテレーション回数は 20 とした。

4.2 SentEval

SentEval[4, 2] は文章埋め込みの評価タスクセットであり、downstream タスク [4] と probing タスク [4] から構成される。ロジスティック回帰により文章ベクトル $\boldsymbol{v}$ を入力とする分類を行う。バッチサイズの 128 とし、最適化は RMSProp で行った。

今回、downstream task から感情分析 (MR, SST2), 質問の種類 (TREC)、商品レビュー (CR)、主観・客観 (SUBJ)、意見極性 (MPQA)、言い換え (MRPC)、論理的含意 (SICK-E) 、意味論的相似性 (SICK-R, STSB) の分類タスクを評価に用いた。Probing task からは, 文章の長さ (SentLen)、文章の特定の単語含有 (WC)、構文木の深さ (TreeDepth)、上位の構成要素 (TopConst)、単語入れ替え (BShift)、主節の時制 (Tense)、主節の主語の個数 (SubjNum)、主節の目的語の個数 (ObjNum)、ランダムな動詞また名詞の入れ替え (SOMO)、主節と従属節の入れ替え (CoordInv) を評価に用いた。

4.3 結果

通常の学習 (vanilla)、敵対的学習 (AdvT), virtual adversarial trainig(VAT) で学習された InferSent による文章埋め込みの SentEval の downstream タスクでのスコアを表 1 に示す。なお、各タスクのスコアは高いものほど性能が良いと評価できる。downstreaming タスクでは、STSB では、vanilla が性能が最も良かったものの、それ以外のタスクでは VAT が vanilla を上回るスコアを示した。その一方で、AdvT では、STSB、MRPC、 TREC で vanilla のスコアを下回る結果を示した。

次に、SentEval の probing タスクでのスコアを表 2 に示す。proving タスクにおいても VAT が vanilla のスコアを上回っていることがわかる。また、downstream task と同様に敵対的学習では, ほとんどのタスクで vanilla よりも高いスコアを見せているものの、CoordInv ではパラメータ ϵ によっては vanilla より劣ったスコアを示すタスクが確認できる。

4.4 考察

実験結果から、VAT によって学習された InferSent が SentEval の多くのタスクにおいて通常の学習よりも高い精度が確認できた。しかし、敵対的学習では、VAT と比較して精度に劣ることが観察された。敵対的正則化を言語処理に適用した研究でも VAT に対して敵対的学習の精度に劣ることが報告されている [7, 10]。この理由については、敵対的学習ではラベルを利用して敵対的摂動を生成するため、敵対的サンプルが VAT に比べてデータに対して過学習を起こしやすい (label leakage) ためであると指摘されている [6, 10]。[7] では、文書分類に対して VAT が有効であることを示しているが、本研究によって直接文書の学習に VAT を適用するのではなく、VAT を用いて学習した文章埋め込みを文書分類に適用した場合でも精度が上がることが示された。本手法では、単語埋め込みが得られるため、より広い応用範囲の可能性が広がる。

5 おわりに

自然言語推論を事前学習とする教師あり文章埋め込み (InferSent) に対して、敵対的学習と VAT を用いて学習の効果を確かめた。二つのうち、VAT で学習された InferSent は通常の学習に比べて SentEval のタスクのほとんどで精度の向上が見られた。このことから、敵対的正則化によって文章埋め込みの性能が向上すると評価できる。既存研究の多くでは、単一の言語処理タスクでの敵対的正則化の効果を検証していることがほとんであったが、この研究では文章埋め込みに対する敵対的正則化の効果を示した。文章埋め込みは様々な言語タスクで特徴量抽出器として利用が可能であるため、より広い範囲で敵対的正則化の応用可能性を広がる。

表 1: SentEval の downstream タスクでのスコア。行が各学習手法、列が各タスクを表す。Vanilla は通常の学習、AdvT は敵対的学習、VAT は virtual adversarial trainig を表す。なお、ϵ は (8)(9) のパラメータである。

tarin/task	ϵ	MR	CR	MPQA	SUBJ	SST2	TREC	MRPC	SICK-R	SICK-E	STSB
Vanilla	-	78.46	81.06	89.65	90.78	81.05	79.8	75.19	88.39	84.74	**76.00**
AdvT	0.25	77.67	82.07	89.77	91.11	81.11	78.6	73.62	88.73	85.83	75.69
AdvT	0.50	79.03	81.91	89.19	90.87	82.26	79.2	74.61	88.59	84.51	74.96
VAT	0.25	**79.77**	**83.47**	**89.93**	**92.38**	81.82	**87.0**	74.61	**88.86**	**86.58**	75.85
VAT	0.50	79.12	84.69	89.71	92.24	**83.31**	88.2	**75.42**	88.48	86.48	74.57

表 2: SentEval の proving タスクでのスコア。行が各学習手法、列が各タスクを表す。Vanilla は通常の学習、AdvT は敵対的学習、VAT は virtual adversarial trainig を表す。

tarin/task	ϵ	SentLEN	WC	TreeDepth	TopConst	BShift	Tense	SubjNum	ObjNum	SOMO	CoordInv
Vanilla	-	67.17	10.29	35.16	62.93	60.29	86.79	84.51	78.27	56.88	63.51
AdvT	0.25	68.79	18.69	36.49	62.17	61.43	86.96	**85.68**	80.05	58.91	59.41
AdvT	0.50	68.41	14.56	32.4	63.76	61.17	86.98	85.03	79.68	58.76	66.12
VAT	0.25	72.1	52.45	38.59	70.2	**61.61**	**87.16**	85.41	**81.17**	59.08	**67.16**
VAT	0.50	**79.32**	**62.8**	**39.02**	**71.62**	61.18	87.08	85.29	80.64	**60.07**	68.2

参考文献

[1] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning. A large annotated corpus for learning natural language inference. In *EMNLP 2015, Conference Proceedings*, pp. 632–642. Association for Computational Linguistics (ACL), 2015.

[2] A. Conneau and D. Kiela. SentEval: An evaluation toolkit for universal sentence representations. *LREC 2018*, pp. 1699–1704, mar 2019.

[3] A. Conneau, D. Kiela, H. Schwenk, L. Barrault, and A. Bordes. Supervised learning of universal sentence representations from natural language inference data. *EMNLP 2017, Proceedings*, pp. 670–680, may 2017.

[4] A. Conneau, G. Kruszewski, G. Lample, L. Barrault, and M. Baroni. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *ACL 2018, Proceedings of the Conference (Long Papers)*, Vol. 1, pp. 2126–2136. ACL, 2018.

[5] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *stat*, 1050:20, 2015.

[6] A. Kurakin, I. J. Goodfellow, and S. Bengio. Adversarial machine learning at scale. *ICLR 2017, Conference Track Proceedings*, nov 2016.

[7] T. Miyato, A. M. Dai, and I. Goodfellow. Adversarial training methods for semi-supervised text classification. In *ICLR 2017, Conference Track Proceedings*. International Conference on Learning Representations, ICLR, 2019.

[8] T. Miyato, S. I. Maeda, M. Koyama, and S. Ishii. Virtual Adversarial Training: A Regularization Method for Supervised and Semi-Supervised Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(8):1979–1993, apr 2019.

[9] J. Pennington, R. Socher, and C. D. Manning. GloVe: Global vectors for word representation. In *EMNLP 2014, Proceedings of the Conference*, pp. 1532–1543. Association for Computational Linguistics (ACL), 2014.

[10] M. Sato, J. Suzuki, and S. Kiyono. Effective Adversarial Regularization for Neural Machine Translation. In *ACL 2019, Proceedings of the Conference*, pp. 204–210, 2019.