トリプレットの Better Negative Sampling による Text Embedding の学習とテキスト分類

満石風斗 [1]　安立健人 [1]　狩野芳伸 [2]
[1] 株式会社マネーフォワード [2] 静岡大学 情報学部
{mitsuishi.futo,adachi.kento}@moneyforward.co.jp
kano@kanolab.net

概要

anchor・positive・negative の各テキストからなるトリプレットを用いた text embedding 学習によるテキスト分類では、1. 対象からのトリプレットのサンプリング、2. これを用いた text embedding の学習、の 2 ステップで学習を行う手法がある。従来のランダムなサンプリングに対して、本研究では anchor テキストと positive テキストの識別をより難しくすることで性能向上が見込める Better Negative Sampling を提案する。livedoor ニュースデータセットから作成した均衡・不均衡データセットと、WRIME-ver2 データセットからサンプリングしたデータセットを用いて性能を検証した。accuracy を評価指標に用い、ランダムなサンプリングに比べて 0.5-1.3 %の精度向上を達成した。

1　はじめに

テキスト分類は言語処理タスクの中でも広く応用されているタスクであり、たとえばレビューの感情分析や英語文法の正誤判定 [1] などに利用されている。近年では BERT[2] などの事前学習モデルを Classifier としてファインチューニングする事例が多くみられるが、データセットに応じた text embedding を獲得し、訓練データとテストデータの text embedding の類似度から分類ラベルを予測する手法も考えられる。text embedding を用いる手法では類似度を確信度として使うことができるなど実用上有利な点がある。

テキスト分類のための text embedding の学習には、anchor テキスト (基準となるテキスト)・positive テキスト (anchor との類似度が大きくなることが期待されるテキスト)・negative テキスト (anchor との類似度が小さくなることが期待されるテキスト) からなるトリプレットを分類の付与されたコーパスから構築し、このトリプレットの集合から text embedding 生成モデルの学習を行う手法が考えられる。

本研究ではこのトリプレット集合の構築方法に着目した。具体的には、ランダムなサンプリングを行うベースラインに対して、negative テキストのサンプリングに工夫を加える Better Negative Sampling を提案する。

トリプレットを用いた学習では、negative テキストに anchor テキストとの分類が難しい hard negative なテキストを用いることがファインチューニングの精度を高めるのに有効である可能性が報告されている [6]。この知見をテキスト分類のための text embedding 学習に適用するために、提案手法である Better Negative Sampling では、negative テキストに「anchor テキストとラベルは異なるが text embedding 間の類似度が最も高いテキスト」を選択する。提案手法を用いることでよりテキスト分類に適した text embedding が得られると期待する。

実験では、ラベルが付与された livedoor ニュースデータセット[1)]から作成した均衡・不均衡データセットおよび WRIME-ver2[2)]からサンプリングしたデータセットに対して、ランダムなサンプリングと Better Negative Sampling を用いて text embedding を学習し比較した結果、Better Negative Sampling を用いたモデルは約 0.5 %の精度向上を達成した。

2　関連研究

2.1　事前学習モデルを用いたテキスト分類

BERT は大規模なデータセットを用いた事前学習とタスクに応じたファインチューニングによって高い精度を示す大規模言語モデルであり [2]、テキス

1）https://www.rondhuit.com/download.html
2）https://github.com/ids-cv/wrime

ト分類を含む多くのタスクに応用されている。

Lewis[7] らはいくつかの事前学習モデルにおいて、データセットに応じた text embedding の学習および転移学習を行うことで、直接 Classifier としてファインチューニングした場合に比べて少ないデータ数で同等の性能でテキスト分類を行えると示唆している。他にも text embedding を用いてテキスト分類を行う手法が提案されているが [9][10]、これらは negative テキスト、すなわちトリプレットを利用していない。

2.2 タスクによらない text embedding の学習

ファインチューニングをしていない BERT からも text embedding は獲得できるが、そのまま用いても性能は低いことが示唆されており [3]、トリプレットサンプルを使用するなどしてより性能高く類似度を計算できるような text embedding を獲得する方法が研究されている [3][4][5]。トリプレットサンプルを用いた学習には negative テキストに hard negative な例を用いることがモデルの停滞を遅らせるのに有効であることが知られており [6]、実際に Ein-Dor ら [11] は Wikipedia のデータセットから同じ記事であるが別の section の文章を hard negative として使用することで、精度の高い text embedding の学習に成功している。

2.3 Multiple Negatives Ranking Loss

トリプレットを使用した学習の際の損失（Loss）には様々な関数が提案されている [3][4][5]。そのひとつである Multiple Negatives Ranking Loss[3] では、トリプレットをいくつか集めて 1 バッチとする。

バッチサイズを K とするとバッチ内には、anchor テキスト集合 A ∋ anchor_i, positive テキスト集合 P ∋ positive_i, negative テキスト集合 N ∋ negative_i ($i = \{1, 2, \cdots, K\}$) が含まれる。図 1 の例のように、各 anchor_i について、positive_i との類似度を大きく、$\text{positive}_j\{j \neq i\}$ および全ての negative との類似度が小さくなるように学習する。そのために anchor_i とバッチ内全ての positive テキストおよび negative テキストの類似度を計算し、positive_i との類似度を正解とみなす Categorical Cross Entropy Loss を最適化する。

すなわち、$\text{candidate}_j(j = 1 \cdots K)$ は positive_j を、$\text{candidate}_j(j = K+1 \cdots 2K)$ は negative_{j-K} を表すとして positive と negative をまとめて candidate と表したとき、各バッチにおいて以下を最適化する。

$$-\sum_i \sum_j l_{i,j} \log \text{Softmax}_{i,j}(cos(model_\theta(\text{anchor}_i), model_\theta(\text{cadidate}_j))$$

ここで、

$$l_{i,j} = \begin{cases} 1 \ (i = j) \\ -1 \ (i \neq j) \end{cases}$$

$$\text{Softmax}_{i,j}(cos(a_i, b_j)) = \frac{e^{cos(a_i, b_i)}}{\sum_j e^{cos(a_i, b_j)}}$$

であり、$\cos(vector_1, vector_2)$ は二つのベクトルのコサイン類似度を、$\text{model}_\theta(t)$ はある時点でのモデルによるテキスト t に対応するベクトルを表す。$(vector_1, vector_2)$ はそれぞれベクトルを表す。

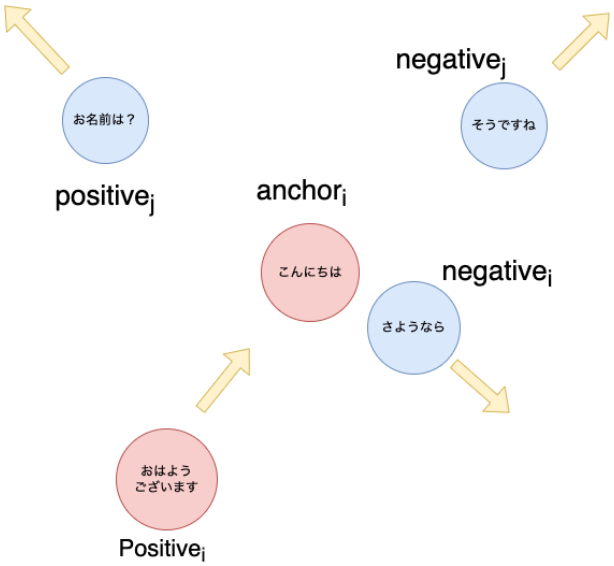

図 1 Multiple Negatives Ranking Loss の概念図: 添え字 i, j は同じバッチ内の異なるトリプレットに対応する。anchor_i と同じラベルに属する positive_i は類似度が高くなり、positive_i 以外の異なるラベルに属する候補テキストの類似度は小さくなるようにモデルを更新する。

3 提案手法

本研究では、タスクによらない text embedding の学習で有効とされている hard negative の利用をテキスト分類のための text embedding の学習に応用するため、negative テキストの選択に工夫を加えた Better Negative Sampling を提案する。

Better Negative Sampling では negative テキストの選択において、「anchor テキストとラベルは異なるが text embedding 間の類似度が最も高いテキスト」を選

3) https://github.com/UKPLab/sentence-transformers

表 1　livedoor コーパスおよび WRIME-ver2 における分類の accuracy スコア。† は各データの助詞・記号を削除したもの。

	livedoor		WRIME-v2	livedoor†		WRIME-v2†
	不均衡	均衡		不均衡	均衡	
BERT Classifier	0.538	0.777	0.444	0.354	0.602	0.405
Text Embedding （random）	0.576	0.777	0.504	0.410	0.688	0.384
Text Embedding （ours）	**0.581**	**0.782**	**0.517**	**0.461**	**0.722**	**0.431**

表 2　1 ラベル内データ数と 1 ラベルの accuracy の相関 (Pearson の積率相関係数)

	BERTClassifier	Text Embedding(random)	Text Embedding（ours）
相関係数	0.409	0.155	0.197

択することでテキスト分類の精度向上を期待する。この選択の際の類似度にはファインチューニング前の事前学習済みモデルから得られる text embedding のコサイン類似度を用いる。

3.1　トリプレットデータセットの作成

ベースラインであるランダムなトリプレットデータセットは元のデータセットから以下のようにサンプリングし作成する。

1. データセット内の全てのサンプルが一度ずつ anchor テキストとなるようにする
2. anchor テキストと同じ分類ラベルのテキストからランダムに positive テキストを抽出する
3. anchor テキストと異なる分類ラベルのテキストからランダムに negative テキストを抽出する

3.2　Better Negative Sampling

提案手法である Better Negative Sampling では以下のようにトリプレットデータセットを作成する。ベースラインのランダムな作成手順 1-2 に加え、

3. 事前学習モデルを用いて、anchor テキストと異なるラベルのテキストと anchor テキストのコサイン類似度を計算する
4. 3 で最も類似度が高いテキストを negative テキストとする

このようにして抽出された negative テキストは hard negative として機能し、text embedding モデルの学習能力を高めると考えられる。

4　実験

4.1　データセット

本研究では、日本語の多クラス分類データセットとして、livedoor ニュースデータセットおよび WRIME-ver2[8] データセットからから作成したデータを用いた。

livedoor ニュース　livedoor ニュースデータセットはニュース記事が 9 つのメディアにラベリングされているデータであり、本研究ではニュース記事の見出しと、記事に紐づけられた 9 種のメディアラベルのみを用いた。ラベル数が多く不均衡なデータセットに対する提案手法の性能も確認するため、livedoor ニュースデータセットから不均衡データセットと均衡データセットを作成した。これらのデータセットの作成方法および統計情報は付録に記した。

WRIME-ver2　WRIME-ver2 はツイートテキストに主観および客観的な 5 段階の感情強度と 8 つの感情の主観および客観的な 4 段階の強さがラベリングされたデータであり、本研究ではツイートテキストと客観的な感情強度を用いて、5 段階の分類タスクとして実験した。計算量を減らすため、WRIME-ver2 からサンプリングしたデータセットを実験に用いた。データセットの作成方法と統計情報を付録に記した。

4.2　モデル

事前学習モデルとして東北大学・乾研究室が公開している whole-word-masking で学習された BERT を使用した[4)]（以下日本語 BERT と表記する）。ベースラインであるランダムなトリプレットデータセット、および Better Negative Sampling で作成したトリプレットデータセットに対して、関連研究 2 で説明した Multiple Negatives Ranking Loss を用いた日本語 BERT のファインチューニングを行い、text embedding モデルを学習した。text embedding モデルを用いたテキスト分類は、テストデータと学習データの text embedding のコサイン類似度をデータごとにそれぞれ計算し、最も類似度が高い学習データのラベルを予測ラベルとした。

4）　https://github.com/cl-tohoku/bert-japanese

表 3 1 ラベル内データ数が 20 以下および 20 以上のラベルに対する accuracy

	20 以下	20 以上
BEETClassifier	0.327	0.579
Text Embedding (random)	0.534	0.590
Text Embedding（ours）	0.517	0.602

5 結果

表 1 に、分類評価結果の accuracy スコアを示す。text embedding を用いた手法は classifier としての学習をおこなった場合に匹敵する精度を示した。

text embedding を用いた手法内で比較すると、Better Negative Sampling を用いたモデルではベースラインであるランダムなサンプリングに比べ 0.5-1.3 %の精度向上が見られた。

6 考察

Better Negative Sampling を用いたモデルは、実験した全てのデータセットにおいてランダムなサンプリングを用いたベースラインモデルに比べて性能向上が見られた。この結果から Better Negative Sampling を用いることで text embedding によるテキスト分類の性能を高めることができることが示唆される。

text embedding によるテキスト分類がサンプルの文構造をどのくらい利用しているのかを調べるため、MeCab[13] の Python ラッパーである fugashi[14](バージョン 1.2.1 IPAdic 辞書) を用いてサンプルから機能語である助詞および句読点などの記号を削除してからラベルの予測を行った結果を表 2(表右側にある † のついたカラム) に示す。Better Negative Sampling がランダムなサンプリングに比べて 3.4-5.1 %の精度改善と、助詞・記号を削除しない場合に比べて精度の向上が大きくなっていた。Better Negative Sampling ではランダムなサンプリングに比べて、文構造をあまり利用しておらず、すなわち内容語を強く学習している可能性がある。

さらに、text embedding を用いたテキスト分類は classifier としての学習に比べ livedoor 均衡データセットでは精度が大きく変わらないのに対し、livedoor 不均衡データセットではランダムなサンプリングで 3.8 %の精度向上が見られた。この結果から text embedding を用いたテキスト分類はランダムサンプリング・Better Negative Sampling を問わず不均衡データに対する頑健性を持つ可能性が考えられる。

表 1 の右側にある、livedoor 不均衡データセットにおけるラベルごとのデータ数と accuracy との相関 (Pearson の積率相関係数) を示している。text embedding を用いたテキスト分類では classifier としてファインチューニングした場合に比べて相関が弱い。この結果から text embedding を用いたテキスト分類はデータ数の少ないラベルに対しても高精度で分類できることがわかる。

さらに、1 ラベル内データ数が 20 未満の場合と 20 以上の場合とでテストデータを分割し、それぞれ各手法の accuracy を調べた結果 3 のようになった。データ数 20 未満のラベルに対する accuracy は、classifier では大きく精度が落ちているのに対し、text embedding を用いた手法では精度がほとんど変わらなかった。この結果から、text embedding を用いた手法は classifier と比較してデータ数が少ないラベルにも頑健に対応できることが示唆された。

7 おわりに

テキスト分類におけるトリプレットのサンプリング手法として、negative テキストを「anchor テキストとラベルが異なるが類似度が最も高いテキスト」とする Better Negative Sampling を提案し、text embedding を用いたテキスト分類の精度が向上することを示した。また、Better Negative Sampling を用いた手法ではランダムなサンプリングに比べ内容語を強く学習している可能性があり、text embedding を用いたテキスト分類は classiffier としてファインチューニングするよりも不均衡データに対し頑健である可能性が示唆された。

今後の展望として、現在の手法では事前学習モデルを用いて hard negative なテキストをサンプリングしているが、1epoch ごとに更新されたモデルを用いてトリプレットをサンプリングし直すなど、サンプリングの方法に更なる改善を試みたい。

参考文献

[1] Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., Bowman, S. R. (2018). GLUE: A multi-task benchmark and analysis platform for natural language understanding. arXiv preprint arXiv:1804.07461.

[2] Devlin, J., Chang, M. W., Lee, K., Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

[3] Reimers, N., Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. arXiv preprint arXiv:1908.10084.

[4] Zhang, D., Li, S. W., Xiao, W., Zhu, H., Nallapati, R., Arnold, A. O., Xiang, B. (2021). Pairwise supervised contrastive learning of sentence representations. arXiv preprint arXiv:2109.05424.

[5] 山岸駿秀, 鈴木貴文, 稲木誓哉. (2020). Wikipedia 記事間の関係を考慮した Triplet Network に基づく BERT の Fine-tuning. In 人工知能学会全国大会論文集 第 34 回 (2020) (pp. 3Rin476-3Rin476). 一般社団法人 人工知能学会.

[6] Hermans, A., Beyer, L., Leibe, B. (2017). In defense of the triplet loss for person re-identification. arXiv preprint arXiv:1703.07737.

[7] Tunstall, L., Reimers, N., Jo, U. E. S., Bates, L., Korat, D., Wasserblat, M., Pereg, O. (2022). Efficient Few-Shot Learning Without Prompts. arXiv

[8] 梶原智之. (2021). WRIME: 主観と客観の感情強度を付与した日本語データセット. 自然言語処理, 28(3), 907-912.

[9] Perone, C. S., Silveira, R., Paula, T. S. (2018). Evaluation of sentence embeddings in downstream and linguistic probing tasks. arXiv preprint arXiv:1806.06259.

[10] Piao, G. (2021, May). Scholarly text classification with sentence BERT and entity embeddings. In Pacific-Asia Conference on Knowledge Discovery and Data Mining (pp. 79-87). Springer, Cham. preprint arXiv:2209.11055.

[11] Ein-Dor, L., Mass, Y., Halfon, A., Venezian, E., Shnayderman, I., Aharonov, R., Slonim, N. (2018, July). Learning thematic similarity metric using triplet networks. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018), Melbourne, Australia (pp. 15-20).

[12] Henderson, M., Al-Rfou, R., Strope, B., Sung, Y. H., Lukács, L., Guo, R., ... Kurzweil, R. (2017). Efficient natural language response suggestion for smart reply. arXiv preprint arXiv:1705.00652.

[13] Kudo, T. (2005). Mecab: Yet another part-of-speech and morphological analyzer. http://mecab. sourceforge. net/.

[14] McCann, P. (2020). fugashi, a tool for tokenizing Japanese in python. arXiv preprint arXiv:2010.06858.

表 4　livedoor ニュースデータセット　サンプル例

	記事タイトル例	メディア
例 1	話題のスマホの駆動時間をのばす！ドコモ REGZA Phone T-01 用大容量バッテリー【モバステ通信】	it-life-hack
例 2	転職で確実に年収アップできる方法を知りたい！」 - 辛口説教部屋 vol.9	livedoor-homme
例 3	おはこんこん、ふぉっくす紺子です！オリジナルグッズがアキバ進出だよ	livedoor-homme

表 5　データセット統計情報

	livedoor(不均衡)	livedoor(均衡)	WRIME-ver2
学習データ数	1863	1863	1863
テストデータ数	234	234	234
ラベル数	109	9	5
1 ラベル平均データ数	19.2	82	419.4
1 ラベル最小データ数	3	233.0	419
1 ラベル最大データ数	45	345	420

A　モデルのパラメータ設定

classifier の学習率は encoder では 5e-5, 分類を行う classifier 層では 1e-4 とした。文章抽出器の学習率は 2e-5 とした。classifier の学習では encoder 層の [CLS] トークンの位置の特徴量を用いてファインチューニングを行った。text embedding 抽出器では最終層の平均をとる mean pooling を用いてファインチューニングを行った。その他のハイパーパラメーターは表 6 に記した。

表 6　学習パラメータ

	optimizer	最大 token 数	batch size	hidden size
BERTClassifier	Adam	128	64	768
Text Embedding	AdamW	128	32	768

B　データセット情報

livedoor 不均衡・均衡データセットは以下のように作成した。各データセットの統計情報は表 5 に記した。

B.1　livedoor 不均衡データセットの作成

1. livedoor ニュースデータセットには 9 種類のメディアがラベリングされている。各メディアにおいて表 4 例 1 例 2 のようにすみかっこ等でサブタイトルが付いてるサンプルについて、サブタイトルを新たなタイトルとし、タイトルテキストからはサブタイトルを削除した。サブタイトルの付いていないサンプルは media 名-other というラベルとした。
2. 表 4 例 3 のように、メディア・ラベル特有の文言がついているサンプルについて、これらの文言を削除した。
3. dokujo-other と分類されたものから、手動で恋愛・結婚・美容　へと分類した。
4. sports-watch-other と分類されたものから、手動でサッカー・野球・オリンピックへと分類した。
5. 各ラベルでサンプル数 2 以下のものは削除した。またサンプル数が 45 以上のものはランダムに 45 サンプルを抽出した。

B.2　livedoor 均衡データセットの作成

livedoor 不均衡データセットのラベルを元の livedoor ニュースデータセットのメディアラベルとした。

B.3　WRIME-ver2 データセットからのサンプリング

5 つの感情強度 (-2, -1, 0, 1, 2) からそれぞれ 419, 419, 419, 420, 420 サンプルをランダムに抽出した。