属性情報を追加した事前学習済みモデルのファインチューニング

笹沢裕一　　岡崎直観
東京工業大学
{yuichi.sasazawa[at]nlp.c, okazaki[at]c}.titech.ac.jp

1 導入

感情分類，要約，文章生成などの幅広い言語処理タスクにおいて，ユーザ ID や製品 ID などの属性情報を活用することでタスクの性能が向上することが報告されている [1, 2, 3]．感情分類の既存研究では，LSTM や CNN モデルに対して注意機構 [4, 5, 6, 7, 8] やメモリーネットワーク [9, 10, 11, 2]，重み行列 [12, 13] などで属性情報を組み込んでいる．

近年，GPT [14], BERT [15], BART [16] などの事前学習済みモデルに対して，様々な言語処理タスクでファインチューニングするアプローチが成功を収めている．そこで，本研究は事前学習済みモデルに属性情報を組み込む手法を探求する．タスクとして感情分類と要約に取り組み，それぞれのタスクで事前学習済みモデルに属性情報を組み込んだモデルを提案する．提案モデルには事前学習済みのパラメータと追加要素に対応する未学習のパラメータの 2 種類が混在する．タスクの学習データでファインチューニングをするときは，後者のパラメータを効率よく学習することが重要であるが，前者のパラメータを更新しすぎると破滅的忘却に陥る．そこで，事前学習モデルと追加要素のパラメータに対して異なる学習率を設定する戦略を提案する．評価実験では，提案手法が感情分類タスクと要約タスクにおいて既存手法の性能を大きく上回り，最高性能の結果を示すことを報告する．また，追加要素のパラメータに対して異なる学習率を設定する戦略は，シンプルでありながら性能向上に大きく貢献することを示す．

2 手法

2.1 感情分類モデル

提案する感情分類モデルの概要を図 1 に示す．標準的な文書分類タスクでは n 単語の単語列

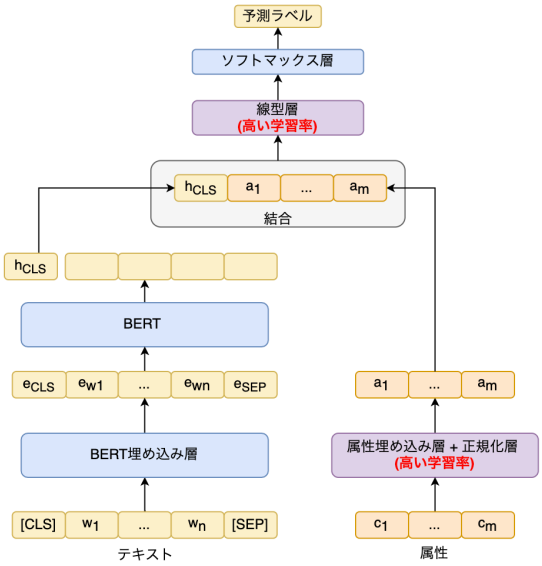

図 1: BERT モデルを使用した感情分類モデル

$x = (w_1, \ldots, w_n)$ を受け取り，ラベル $y \in \mathcal{Y}$（$\mathcal{Y}$ は l 種類のラベルから構成されるラベル集合）を予測する．本タスクでは m 種類のカテゴリ属性 $(c_1, \ldots c_m)$ も与えられる．本研究の感情分類データセットではユーザ ID と製品 ID の 2 種類のカテゴリ属性が与えられるため，$m = 2$ である．

我々は感情分類タスクの事前学習済みモデルとして BERT モデル [15] を使用する．BERT モデルは一連のトークン $([\text{CLS}], w_1, \ldots, w_n, [\text{SEP}])$ を入力として受け取り，各入力トークンに対応する隠れ状態 $[h_{\text{CLS}}, h_1, \ldots, h_n, h_{\text{SEP}}]$ を生成する．既存研究に倣い，$h_{\text{CLS}} \in \mathbb{R}^d$ を入力の分散表現として使用する（d は次元数）．属性情報をモデルに考慮させる様々な手法を比較したところ，テキスト分散表現と属性埋め込み表現を結合する手法が最も優れた性能を示すことが分かった．つまり，提案モデルではテキストの予測ラベルを次式により算出する．

$$a_k = \text{LayerNorm}(\text{emb}(c_k)) \quad (1)$$

$$y = \text{Softmax}(W[h_{\text{CLS}}; a_1; \ldots; a_m] + b) \quad (2)$$

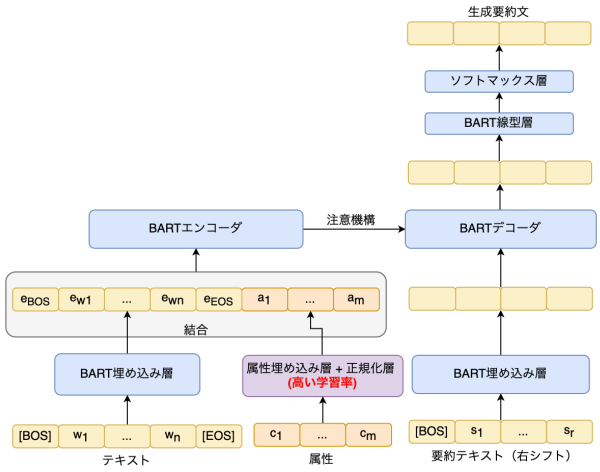

図 2: BART モデルを使用した要約モデル

emb は埋め込み層，LayerNorm は β, γ をパラメータとして持つ Layer Normalization 層による正規化処理であり $a_k \in \mathbb{R}^{d_a}$ である．$W \in \mathbb{R}^{l\times(d+d_a m)}$ と $b \in \mathbb{R}^l$ は線形層のパラメータ，Softmax はソフトマックス関数，$[.;.]$ はベクトルの結合である。我々の提案モデルではテキスト分散表現 h_{CLS} と各属性の埋め込み表現 $[a_1, \ldots, a_m]$ の結合を後続の線形層に与えることで最終的な予測ベクトル y を計算する．

2.2 要約モデル

要約モデルの概要を図 2 に示す．要約タスクでは n 単語の単語列 $x = (w_1, \ldots, w_n)$ を受け取り，要約単語列を予測する．テキスト分類と同様に m 種類のカテゴリ属性 $(c_1, \ldots c_m)$ が与えられ，$m = 2$ である．

本研究では要約タスクの事前学習モデルとして BART モデル [16] を使用する．入力テキストの埋め込み表現を $[e_{\mathrm{BOS}}, e_{w_1}, \ldots, e_{w_n}, e_{\mathrm{EOS}}] \in \mathbb{R}^{d\times(n+2)}$ とする（e_w は単語 w の埋め込み表現である）．通常の要約タスクではこのテキスト埋め込み表現を BART エンコーダに入力することにより要約モデルを学習する．提案モデルでは，属性情報を考慮するためにテキスト埋め込みと属性埋め込みをエンコーダに入力する．つまり，感情分類モデルと同様の手法で獲得した属性埋め込み $[a_1, \ldots, a_m] \in \mathbb{R}^{d\times m}$ を使用し，$[e_{\mathrm{BOS}}, e_{w_1}, \ldots, e_{w_n}, e_{\mathrm{EOS}}, a_1, \ldots, a_m] \in \mathbb{R}^{d\times(n+2+m)}$ を BART エンコーダに入力する．

2.3 追加事前学習

既存研究はターゲットコーパス上での追加の事前学習により，BERT モデルを用いたテキスト分類タスクの性能がさらに向上したと報告している [17, 18]．要約タスクにおいても事前学習済みモデルの学習において追加の事前学習がモデルの性能の向上に貢献すると報告されている [19]. これらの研究に倣い，ファインチューニングの前にそれぞれのタスクの訓練データを用いてマスクトークンの予測による事前学習を行う．

2.4 複数の学習率の使用

我々のモデルは事前学習済みモデルのパラメータと，追加要素に含まれる未学習のパラメータから構成される．感情分類タスクでは $(\mathrm{emb}, \beta, \gamma, W, b)$ が追加要素に含まれる未学習のパラメータである．多くの機械学習モデルではモデル全体に一律の学習率（オプティマイザに与える初期学習率）を設定して訓練を行っている．しかし，事前学習済みのパラメータと追加要素のパラメータの両方に対して一律の学習率を用いて学習を行うことは不適切である可能性がある．つまり，高い学習率を用いてモデル全体を訓練した場合，事前学習済みモデルは壊滅的忘却に陥る．一方，低い学習率を用いた場合は追加要素のパラメータを十分に訓練できない．

この問題に対処するために，図 1, 2 に示すように事前学習済みモデルと追加要素のパラメータに対して異なる学習率を使用する．つまり，我々は追加要素には 1×10^{-3} のような高い学習率を使用し，事前学習モデルには 1×10^{-5} のような低い学習率を使用する．後に実験で示すように，2 種類の学習率を使用するこの戦略は事前学習済みモデルに追加要素を組み込んだモデルの性能を大きく改善する．

パラメータごとに異なる学習率を用いるというアイデア自体は様々な研究で提案されている．Adagrad [20] や Adam [21] などのオプティマイザはそれぞれのパラメータが受けた勾配に基づいて個別の学習率を設定しているが，一律の初期学習率ではモデルの最適化に限界がある．Sun ら [17] は BERT モデルの層ごとに複数の学習率を使用する戦略を提案しているが，追加要素に対しては BERT モデルと等しい低い学習率を用いている．

3 実験

3.1 実験設定

ハイパーパラメータ設定，既存手法の説明，データセットの統計を含めたより詳細な設定は付録 A を参照されたい．それぞれのモデルにおいて追加事前

表 1: 感情分類タスクの実験結果

手法	IMDB		Yelp 2013		Yelp 2014	
	正解率	RMSE	正解率	RMSE	正解率	RMSE
既存手法（属性あり）						
CMA [5]	54.0	1.191	66.4	0.677	67.6	0.637
RRP-UPM [11]	56.2	1.174	69.0	0.629	69.1	0.621
MOCA [8]	56.9	1.060	70.6	0.596	71.8	0.578
BERT（属性なし）						
BERT	54.0	1.112	70.5	0.578	70.3	0.580
BERT + 追加事前学習	54.7	1.085	70.9	0.568	70.9	0.568
BERT + 追加事前学習 + 複数の学習率	55.0	1.078	70.8	0.569	71.0	0.568
BERT（属性あり）						
BERT	56.5	1.096	72.2	0.561	72.2	0.565
BERT + 追加事前学習	57.2	1.059	72.5	0.553	73.0	0.550
BERT + 追加事前学習 + 複数の学習率	**60.4**	**0.987**	**74.2**	**0.537**	**74.3**	**0.533**

表 2: 要約タスクの実験結果

手法	R1	R2	RL
既存手法（属性あり）			
AttrEnc [2]	16.6	5.6	16.3
MemAttr [2]	18.0	6.8	17.8
BART（属性なし）			
BART	14.8	5.6	14.4
BART+追加事前学習	15.2	5.7	14.9
BART（属性あり）			
BART	18.3	7.3	17.8
BART+追加事前学習	19.0	7.5	18.4
BART+追加事前学習+複数の学習率	**20.8**	**8.9**	**20.3**

学習と複数の学習率を使用する手法を試し，モデルの性能に対する影響を確認する．

感情分類　3 種類のレビューデータセット：IMDB，Yelp 2013，Yelp2014 [1] を用いた．分類のターゲットはレビュースコアであり，IMDB は 1〜10 の 10 種類，Yelp では 1〜5 の 5 種類である．カテゴリ属性としてレビューを記載したユーザーの ID と，レビューの対象となる製品の ID を使用した．事前学習モデルは BERT モデルとして $\text{RoBERTa}_{\text{BASE}}$ [22] を使用した．既存研究と同様に分類の正解率と二乗平均平方根誤差 (RMSE) を報告する．BERT（属性なし）はテキストのみを BERT モデルに入力して分類を行うモデルであり，このモデルにおいても複数の学習率を試した．つまり，追加要素である線形層の学習率を，BERT モデルに対する学習率よりも高い値に設定した．

要約　データセットは Amazon のレビューデータ [2] を使用した．要約のターゲットはレビューのタイトルである．感情分類タスクと同様に，ユーザ ID と製品 ID がカテゴリ属性として与えられる．事前学習モデルは $\text{BART}_{\text{BASE}}$ を使用した．既存研究と同様に Rouge-1, Rouge-2, Rouge-L を報告する．

3.2　実験結果

表 1 は感情分類タスクにおける実験結果である．我々の提案した BERT ベースの手法が既存手法の結果を大きく上回り，最高性能の結果を達成している．また，複数の学習率を使用する戦略によってモデルの性能が大きく向上している．一方，属性情報を使用せずにテキストのみを使用して BERT モデルで分類を行う場合は，複数の学習率を設定してもモデルの性能をほとんど向上させることができない．この結果より，複数の学習率を使用する戦略は属性情報を事前学習済みモデルと組み合わせて使用する際に特に有用であることが分かる．また，追加事前学習は属性情報の有無に関わらずモデルの性能を向上させていることが確認できる．

表 2 は要約タスクにおける実験結果である．要約タスクにおいても事前学習済みモデルをベースとした手法が最高精度を達成している．また，感情分類タスクと同様に，複数の学習率を使用する手法と追加事前学習も性能の向上に寄与していることを確認できる．特に，複数の学習率を設定することで Rouge スコアの大幅な上昇が見られる．

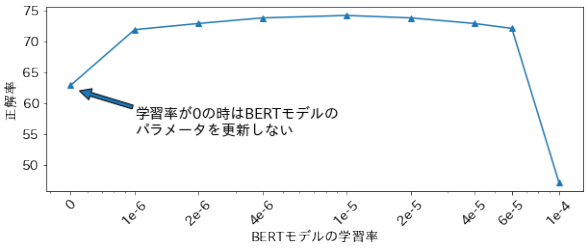

図 3: BERT モデルの学習率と正解率の関係

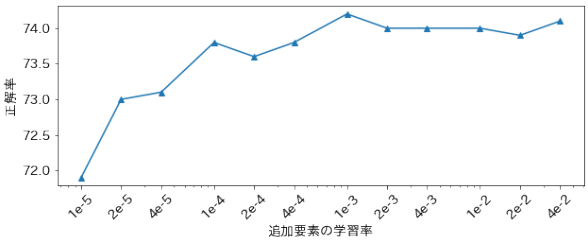

図 4: 追加要素の学習率と正解率の関係

3.3 学習率と正解率の関係

図 3 は感情分類データセットの Yelp2013 において，追加要素の学習率を 1×10^{-3} に固定し，事前学習済み BERT の学習率のみを変化させた時の正解率の変化を示したものである．学習率を高くし過ぎると破滅的忘却もしくはパラメータの発散が発生し，正解率が低下する．一方，学習率を低くし過ぎた場合もパラメータを十分に学習できず，正解率が低下する．図 4 は BERT の学習率を 1×10^{-5} に固定し，追加要素の学習率のみを変化させた時の正解率の変化を示したものである．追加要素の学習率が低いと十分に学習が行われず，正解率が低下することが分かる．これらの結果より，事前学習済みのパラメータと追加要素のパラメータのそれぞれに対して学習に最適な学習率が存在し，複数の学習率の使用がモデルの訓練において重要であることが分かる．

3.4 パラメータの変化量

学習率の設定によるモデルのパラメータに対する影響を確認する．事前学習済みモデルの BERT と追加要素のそれぞれに対して，パラメータの初期状態からの変化量の絶対値の平均値 $\delta_t = \frac{1}{|P|} \sum_{p \in P} \|p_t - p_0\|$ を一定ステップごとに記録する．ただし P は対象とする要素の全てのパラメータ集合，p_t は訓練中の t ステップ目におけるパラメータ p の値である．一律の学習率を使用する設定ではモデル全体の学習率を 1×10^{-5} に設定した．複数の学習率を使用する設定では，BERT モデルの学習率

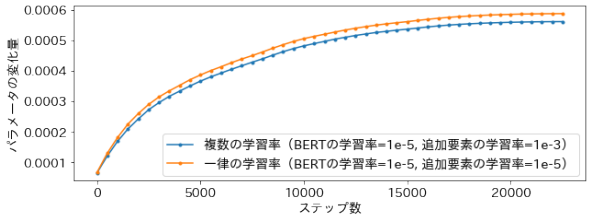

図 5: BERT モデルのパラメータ変化量

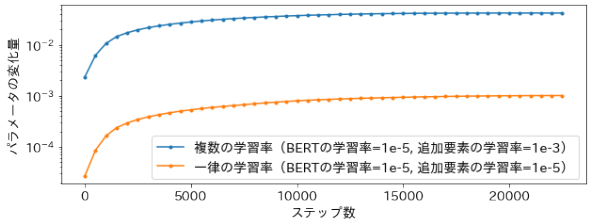

図 6: 追加要素のパラメータ変化量

を 1×10^{-5} に，追加要素の学習率を 1×10^{-3} に設定した．つまり，BERT モデルの学習率は両方の設定で固定であり，追加要素の学習率のみが異なる．

図 5 は Yelp2013 データセットにおける BERT モデルのパラメータ変化量であり，BERT モデルに対する学習率が一定の場合は追加要素の学習率に関わらず，パラメータの変化量はほとんど変化しないことが分かる．一方，図 6 は追加要素のパラメータ変化量であり，追加要素の学習率を低く設定するとパラメータの変化量が小さくなることが分かる．つまり，一律の低い学習率では追加要素のパラメータを十分に最適化できず，それがモデルの性能向上の足枷になると推測される．

4 結論・今後の課題

本研究では事前学習済みモデルに対して属性情報を組み込むための手法を探求した．感情分類タスクと要約タスクのそれぞれにおいて属性情報を組み込んだ提案手法は，既存手法の性能を大きく上回り，最高性能を達成した．また，追加要素のパラメータに対して異なる学習率を設定する戦略は，性能向上に大きく寄与することを示した．

今後は，属性情報とテキストの相互作用を考慮できるさらに性能の高いモデルの検討し，他の言語処理タスクにおいても事前学習済みモデルと複数の学習率を用いた手法が性能の向上に寄与するか実験を重ね，性能向上の要因を更に分析したい．

謝辞

本研究は JSPS 科研費 19H01118 の助成を受けた．

参考文献

[1]Duyu Tang, Bing Qin, and Ting Liu. Learning semantic representations of users and products for document level sentiment classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 1014–1023, 2015.

[2]Hui Liu and Xiaojun Wan. Neural review summarization leveraging user and product information. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19, p. 2389–2392, 2019.

[3]Li Dong, Shaohan Huang, Furu Wei, Mirella Lapata, Ming Zhou, and Ke Xu. Learning to generate product reviews from attributes. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pp. 623–632, 2017.

[4]Huimin Chen, Maosong Sun, Cunchao Tu, Yankai Lin, and Zhiyuan Liu. Neural sentiment classification with user and product attention. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1650–1659, 2016.

[5]Dehong Ma, Sujian Li, Xiaodong Zhang, Houfeng Wang, and Xu Sun. Cascading multiway attentions for document-level sentiment classification. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 634–643, 2017.

[6]Reinald Kim Amplayo, Jihyeok Kim, Sua Sung, and Seungwon Hwang. Cold-start aware user and product attention for sentiment classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2535–2544, 2018.

[7]Zhen Wu, Xin-Yu Dai, Cunyan Yin, Shujian Huang, and Jiajun Chen. Improving review representations with user attention and product attention for sentiment classification. In *Proceedings of the Thirty Second AAAI Conference on Artificial Intelligence*, pp. 5989–5996, 2018.

[8]J. Zhang and C. Chow. Moca: Multi-objective, collaborative, and attentive sentiment analysis. *IEEE Access*, Vol. 7, pp. 10927–10936, 2019.

[9]Zi-Yi Dou. Capturing user and product information for document level sentiment analysis with deep memory network. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 521–526, 2017.

[10]Yunfei Long, Mingyu Ma, Qin Lu, Rong Xiang, and Chu-Ren Huang. Dual memory network model for biased product review classification. In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pp. 140–148, 2018.

[11]Zhigang Yuan, Fangzhao Wu, Junxin Liu, Chuhan Wu, Yongfeng Huang, and Xing Xie. Neural review rating prediction with user and product memory. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, p. 2341–2344, 2019.

[12]Jihyeok Kim, Reinald Kim Amplayo, Kyungjae Lee, Sua Sung, Minji Seo, and Seung-won Hwang. Categorical metadata representation for customized text classification. *Transactions of the Association for Computational Linguistics*, Vol. 7, pp. 201–215, 2019.

[13]Reinald Kim Amplayo. Rethinking attribute representation and injection for sentiment classification. 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. 5602–5613, 2019.

[14]Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. Technical report, 2018.

[15]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, Volume 1 (Long and Short Papers)*, pp. 4171–4186, 2019.

[16]Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7871–7880, 2020.

[17]Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. How to fine-tune bert for text classification? In Maosong Sun, Xuanjing Huang, Heng Ji, Zhiyuan Liu, and Yang Liu, editors, *Chinese Computational Linguistics*, pp. 194–206, 2019.

[18]Alexander Rietzler, Sebastian Stabinger, Paul Opitz, and Stefan Engl. Adapt or get left behind: Domain adaptation through BERT language model finetuning for aspect-target sentiment classification. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pp. 4933–4941, 2020.

[19]Xingxing Zhang, Furu Wei, and Ming Zhou. HIBERT: Document level pre-training of hierarchical bidirectional transformers for document summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 5059–5069, 2019.

[20]John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, Vol. 12, No. 61, pp. 2121–2159, 2011.

[21]Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *International Conference on Learning Representations*, 2015.

[22]Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint*, Vol. arXiv:1907.11692, p. (15 pages), 2019.

[23]Nitish Shirish Keskar, Bryan McCann, Lav R. Varshney, Caiming Xiong, and Richard Socher. CTRL: A conditional transformer language model for controllable generation. *arXiv preprint*, Vol. arXiv:1909.05858, p. (18 pages), 2019.

表 3: データセットの統計

データセット	#訓練	#開発	#評価	#ユーザ	#製品
感情分類					
IMDB	67,426	8,381	9,112	1,310	1,635
Yelp 2013	62,522	7,773	8,671	1,631	1,633
Yelp 2014	183,019	22,745	25,399	4,818	4,194
要約					
Amazon	157,205	5,000	5,000	3,080	3,329

A 詳細な実験設定

データセットの統計を表 3 に示す．オプティマイザとして Adam オプティマイザ [21] を使用し，$\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-6}$, L2 正則化係数は 0.01 である．学習率は訓練ステップ数の最初の 6%でウォームアップさせ，その後線形に減衰させた．ドロップアウト率は 0.1，属性埋め込みを含めた全ての次元数は 768 である．つまり，$d = d_a = 768$ である． ファインチューニングでは両タスクにおいてバッチサイズを 8，入力テキストの最大トークン数を 512 に設定した．入力テキストが 510 トークン [1] より長い場合は Sun ら [17] の手法に倣い，最初の 128 トークンと最後の 382 トークンを抽出し，これを入力テキストとして使用した．

感情分類 BERT モデルの学習率は 1×10^{-5}，追加要素の学習率は 1×10^{-3} を設定し，一律な学習率を使用する設定では 2×10^{-5} を使用した．追加事前学習ではバッチサイズを 32，学習率を 5×10^{-5}，入力テキストの最大トークン数を 128，エポック数を 20 に設定した．エポック数は $\{2, 3, 4, 5\}$ の候補の中から最適な値を探索し，IMDB, Yelp 2013 では 3 回，Yelp 2014 では 2 回のエポック数を使用した．以下に既存手法を示す．

CMA [5] はテキストエンコーダとして階層的な LSTM を使用しており，2 段階の注意機構を使用してカテゴリ属性を組み込んでいる．

RRP-UPM [11] はテキストエンコーダとして LSTM と CNN を組み合わせた階層的なモデルを使用している．また，メモリーネットワーク機構を使用して獲得した属性埋め込み表現を注意機構によってエンコーダに組み込んでいる．

MOCA [8] は注意機構と多層パーセプトロンを使用し，LSTM モデルによって獲得されたテキスト分散表現と属性埋め込みの相互作用を捉えている．また，損失関数としてレビュースコアの分類による損失と回帰による損失の和を使用している．

1) 512 から入力テキストに付与される 2 つの特殊トークンの数を引いた値である．

表 4: 各追加要素に対する高い学習率による影響

高い学習率を設定する要素			正解率	RMSE
埋め込み層	正規化層	線形層		
			71.7	0.563
✓			72.3	0.568
	✓		72.0	0.560
		✓	72.8	0.552
✓	✓		73.8	0.542
✓		✓	73.5	0.544
	✓	✓	72.9	0.551
✓	✓	✓	**74.2**	**0.537**

要約 BART モデルの学習率は 2×10^{-5}，追加要素の学習率は 4×10^{-3} を設定し，一律な学習率を使用する設定では 1×10^{-4} を使用した．追加事前学習ではバッチサイズを 16，学習率を 2×10^{-5}，最大トークン数を 128，エポック数を 20 に設定した．エポック数は $\{2, 3, 4, 5\}$ の候補の中から最適な値を探索し，2 回のエポック数を使用した．要約生成時のビームサイズは 2，生成要約長に対するペナルティ (length penalty) は 1.3，同じ単語の繰り返しに対するペナルティ (repetition penalty) [23] は 3.0 である．以下に既存手法を示す．

AttrEnc [2] は属性埋め込みとテキスト埋め込みの結合を LSTM エンコーダに入力することで属性情報を組み込んでいる．

MemAttr [2] は属性情報をメモリーネットワークで埋め込み，Pointer-generator 機構により属性埋め込みを LSTM モデルに組み込んでいる．

B 追加要素に対する高い学習率の使用による影響

表 4 は感情分類データセットの Yelp2013 において，追加要素（属性埋め込み層，正規化層，線型層）の一部に対して高い学習率 (1×10^{-3}), その他の要素に対しては低い学習率 (1×10^{-5}) を設定した時の性能を示したものである．全ての追加要素について，高い学習率を使用することで正解率が向上することが確認できる．しかし，一つの要素の学習率を高くしただけでは正解率の向上は小さく，全ての追加要素に対する学習率を高くした場合の正解率が最も高い．この結果より，全ての追加要素のパラメータは高い学習率で訓練する必要があり，それによってモデル全体の十分な訓練が可能であることが分かる．