# 教師有り学習モデルと大規模言語モデルを組み合わせた低評価レビューを考慮したレビュー文書の評価値推定

竹尾 匡貴 [1]　嶋田 和孝 [1]
[1] 九州工業大学大学院
takeo.masaki215@mail.kyutech.jp　shimada@ai.kyutech.ac.jp

## 概要

本研究では，教師有り学習モデルと大規模言語モデルを組み合わせた，レビュー文書の評価値推定に取り組む．低評価なレビュー文書は，高評価なレビュー文書に比べて少数派であることが考えられる．そのような場合，教師有り学習モデルによる分類では，特に少数派のクラスの精度が悪化する傾向がある．そこで，低評価レビュー文書の推定にのみ，追加学習無くタスクを解ける大規模言語モデルを導入する．実験の結果から，教師有り学習モデルやデータ拡張手法よりも少数派クラスの精度が改善したことを確認した．また，大規模言語モデルのみの推定よりも，全体の精度バランスが良いことも確認した．

## 1　はじめに

レビュー文書は，製品やサービスを利用した消費者から見た，良くなかった点や改善すべき点が記載されている．製品やサービスの提供者にとって，そのようなレビュー文書に含まれる記述から改善すべき点を見つけ出すことは重要である．一般に，良くなかった点や改善すべき点は低い評価値のレビュー文書に含まれることが多い．一方で，そのような極端に低い評価値を持つレビュー文書は全体の分布の中では少なく，中程度やそれ以上の評価値が付与されたレビューが多数を占めることも多くある．その結果，低い評価値が付与されたレビュー文書は全体の中で少数派になることが多い．

レビュー文書の評価値を推定する手法として，教師有り学習モデルに基づく手法は枚挙に暇がない [1][2][3][4]．教師有り学習においては，データが多いクラスでは文章とそのクラスの関係を十分に学習できる一方，データが少ないクラスでは相対的にモデルの性能が落ちてしまう傾向がある．前述のように，低評価レビューのデータ数はそれ以外のレビュー文書に比べて少ない場合が多いため，全体として各クラスの分布に偏りがあるデータになりやすい．このような条件下で単純な教師有り学習を適用すると，低評価レビューのみならず，全体の分類精度の低下を引き起こす．この問題を解消するには，すべてのクラスについて，レビュー文書が均一に収録されている学習データを用意することが望ましい．しかし，偏りがない学習データを用意することは非常に困難であり，偏った学習データを使用せざるを得ない場合も多い．

偏った学習データを扱う際に少数派クラスの精度を維持する手法として，アンダーサンプリングやオーバーサンプリング，データ拡張がある．しかし，アンダーサンプリングでは総データ数が減少するという問題があり，オーバーサンプリングでは少数派クラスにおける過学習が助長されるという問題が残る [5]．データ拡張アプローチにおいても，単純な単語置換では近年主流になっている transformer ベースのモデルに対して有効でないことや [6]，拡張したデータの質がオリジナルデータより劣る可能性があること [7] が問題点として挙げられる．よって，これらのアプローチの単純な適用は，必ずしも適切な解法ではない場合がある．

近年では，自然言語による指示でタスクを解くことができる大規模言語モデル（Large Language Model: LLM）が広く使用されている．LLM は，タスクを解く際に特段の学習を必要としないため，本質的に，処理対象となるデータのクラス分布や偏りに影響されない．つまり，データ数が少ないと考えられる低評価レビュー文書の分類にも有効だと思われる．一方で，前述のように，製品やサービスの改善のために低評価レビューを正しく抽出できることは重要であるが，レビューの評価値推定という全体的な視点で見た場合，当然ながらそれ以外の評価値


This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

についても良い精度で推定できることが望ましい．しかしながら，Bai ら [8] らや Šmíd ら [9] による報告では，単純な LLM による文書分類の精度は，全体の精度で教師有り学習モデルに劣るとされている．これより，LLM を単純に用いるだけでは，評価値推定というタスク全体の最適な解法にはなり得ない．

そこで本研究では，追加学習無くタスクを解くことができる大規模言語モデルと，教師有り学習モデルを組み合わせた評価値推定手法を提案する．具体的には，データ数が少ない低評価レビューの評価値推定には LLM を，それ以外のレビュー文書の評価値推定には教師有り学習済みモデルを用いる統合的な枠組みを提案する．

## 2 関連研究

レビュー文書の分類は，自然言語処理分野における重要なタスクの一つである [10][11]．しかし，1 章で述べたとおり，分布が偏ったデータに単純に教師あり学習を適用するのは必ずしも最良な手ではない．この問題に対応するために，教師有り学習モデルの学習に用いる損失関数を変更するアプローチが用いられる [5]．例えば，Li ら [12] は，各クラスの予測確率が 50%だけあれば最終的な分類結果として出力可能な点に着目し，少数派クラスの予測確率を無理に 100%に近づけるのではなく，50%に近づけるような損失関数（self-Adjustment Dice Loss: ADL）を提案している．また Lin ら [13] は，通常用いられる Cross Entropy（CE）に対して，正解クラスの予測確率から 1 を引いた数を重みとして付与する損失関数（Focal Loss: FL）を提案している．他にも Nishiyama ら [14] は，注目したいクラスの Recall を改善するため，注目したいクラスの Embedding の分散を抑えるような損失関数（CAMRI Loss）を提案している．Adel ら [15] も，Santos ら [16] が提案した Ranking Loss（RL）を用いて，クラス分布が極端に偏った分類タスクに取り組んでいる．しかし，損失関数の変更だけでは十分に精度が改善されない場合も起こりうる．

## 3 提案手法

本研究では，学習データに収録されているクラス分布の偏りに影響されないレビュー文書の評価値推定手法を提案する．

提案手法の概要を図 1 に示す．まず，レビュー文書を LLM に入力し，LLM による評価値推定を実施する．LLM が推定した評価値を暫定的な評価値とする．この暫定的な評価値の値が事前に設定された少数派クラスの値であれば，その値をそのまま最終的な評価値として扱うこととする．例えば，ここで評価値が 1〜3 の値と取る設定だと仮定し，評価値 1 が少数派，評価値 2 と 3 がそうでない（少数派以外）とする．LLM があるレビューの評価値を 1 と見積もった場合は，（実際の評価値がいくらかに関係なく）最終的な評価値が 1 と確定することを意味する．

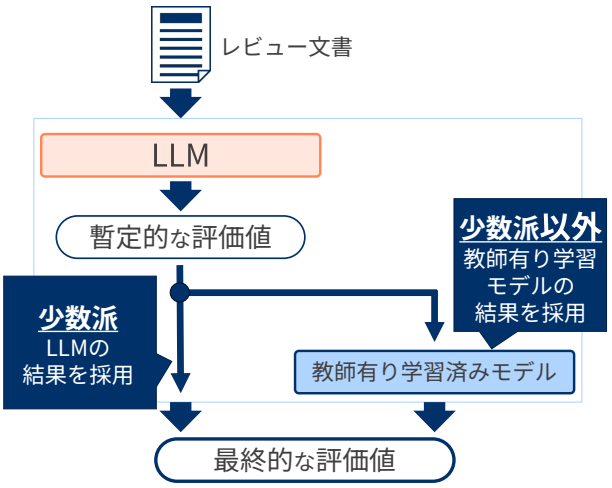


**図 1** 提案手法．LLM が多数派クラスの評価値と推定した場合，教師有り学習モデルを用いて再推定し，その結果を最終的な結果とする．

暫定的な評価値が少数派クラスでない場合は，教師有り学習モデルを利用して，評価値を推定する．このとき，教師有り学習に利用するデータは，少数派以外のデータのみとする．すなわち，先ほどの例で考えると，教師有り学習モデルは，評価値 2 と 3 のデータのみを利用して学習され，出力結果も評価値 2 か 3 しか取らないことを意味する．

以上の手続きにより，学習データで極端に少ない少数派事例に対しては LLM による推定で精度を保ち，十分に学習できそうな少数派以外のデータには教師有り学習モデルを適用し，全体としてバランスのよい評価値推定モデルを実現する．さらに，教師有り学習モデルの学習の際に，2 章で説明したいくつかの損失関数の変更に基づくアプローチも適用し，その有効性を検証する．これは，先述の例でいうと，評価値 2 と評価値 3 のデータが均等である保証はなく，少数派ほどではないにしても偏りが生じる可能性は否定できないためである．


This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

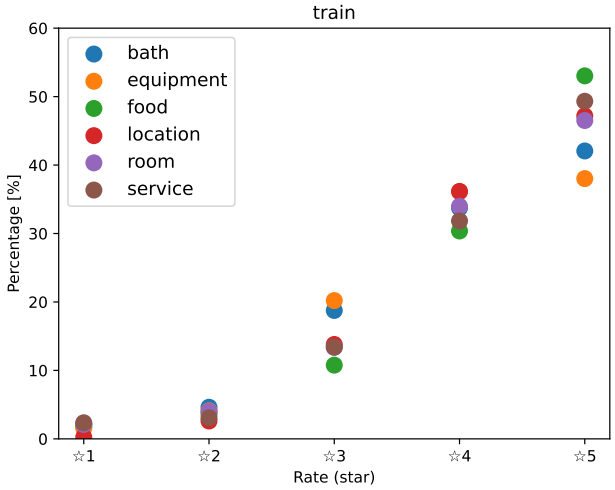


**図 2**　学習データにおける各評価値の割合．6 観点の全てで，評価値が高いほど全体に占める割合が高い．

## 4　実験

ここでは，提案手法の有効性を検証するため，レビューデータを用いた実験を行う．

### 4.1　データセット

実験には，「楽天トラベル: レビューアスペクト・センチメントタグ付きコーパス」[17] を使用する．このデータには，宿泊施設に対する実際に宿泊したユーザからのレビュー文書が 10937 件収録されている．評価値（☆）は，宿泊施設の 6 観点に対してユーザが 1〜5 の 5 段階で付与したものである．学習: 開発: 評価が 3:1:1 となるよう分割する．

図 2 に，学習データにおける 6 観点の評価値分布を示す．図より，☆ 1 が極端に少なくないことがわかる．このデータでは，評価値が上がっていくほどその割合は多くなっていき，☆ 5 は全体の 40%〜50%程度を占めている．以降，☆ 1 と☆ 2 を少数派，☆ 3 から☆ 5 を便宜上多数派と呼ぶ．

### 4.2　実験設定

手法の有効性は，推定評価値と正解評価値の平方平均二乗誤差（Root Mean Square Error: RMSE）で測る．RMSE の micro 平均と macro 平均を算出し，手法ごとの有効性を議論する．micro 平均は，各クラスの事例数に影響を受ける．したがって，少数派クラスの精度改善度合いが現れにくい．そこで，少数派クラスの精度とそれ以外のクラスの精度を同列に評価するために，各クラスの RMSE の値の単純な平均となる macro 平均も算出する．

教師有り学習モデルには BERT [18] を採用する．2 章で述べた通り，偏ったデータを学習する際は適した損失関数に変更することが多い．それら損失関数を BERT に適用する．各損失関数の式やハイパーパラメータは付録 A，B に示す．LLM には，GPT-4o-mini[1] を採用する．GPT-4o-mini へのプロンプト等は付録 C に示す．また，3 章で述べたとおり，BERT と LLM を組み合わせる提案手法（COMBI）においても，BERT の損失関数を変更して実験する．これは，今回多数派として扱っている☆ 3〜☆ 5 にも偏りがあるためである．

**BERT**$_{MSE}$　最も基本的なベースラインモデル．損失関数に Mean Square Error（MSE）を採用し，回帰問題で解く．

**BERT**$_{WCE}$　BERT の損失関数を Weighted Cross Entropy（WCE）に変更したモデル．少数派クラスに重み α を付して学習する．α は [12] に倣う．

**BERT**$_{FL}$　BERT の損失関数を Focal Loss（FL）[13] に変更したモデル．

**BERT**$_{DL}$　BERT の損失関数を Dice Loss（DL）[19] に変更したモデル．

**BERT**$_{ADL}$　BERT の損失関数を self-Adjustment Dice Loss（ADL）[12] に変更したモデル．

**BERT**$_{RL}$　BERT の損失関数を Ranking Loss（RL）[16] に変更したモデル．

**LLM**　全データを LLM が解く．

**COMBI**$_{MSE}$　LLM と MSE を損失関数として学習した BERT を組み合わせる．

**COMBI**$_{WCE}$　LLM と WCE を損失関数として学習した BERT を組み合わせる．

**COMBI**$_{FL}$　LLM と FL を損失関数として学習した BERT を組み合わせる．

**COMBI**$_{ADL}$　LLM と ADL を損失関数として学習した BERT を組み合わせる．

### 4.3　実験結果・考察

6 観点それぞれで評価値推定した結果の平均の数値を表 1 に示す．BERT 単体の手法（BERT$_{***}$）はいずれも少数派の精度が悪く，多数派の精度が良い．これは，教師有り学習モデルの性能が学習データの割合に影響されたからだといえる．多数派の精度が良いため，micro 平均は 0.8 程度と良い傾向にある．一方で，少数派の精度が非常に悪いため，macro 平

1）　https://platform.openai.com/docs/models#gpt-4o-mini


This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

**表 1**　6 観点平均の結果．各評価値で最良の手法は「**太字+下線**」，2 番目に良い手法は「**太字**」で示している．

| 6 観点平均 | | 組み合わせ 無し | | | | | | | 組み合わせ 有り（提案手法） | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | $\text{BERT}_{MSE}$ | $\text{BERT}_{WCE}$ | $\text{BERT}_{FL}$ | $\text{BERT}_{DL}$ | $\text{BERT}_{ADL}$ | $\text{BERT}_{RL}$ | LLM | $\text{COMBI}_{MSE}$ | $\text{COMBI}_{WCE}$ | $\text{COMBI}_{FL}$ | $\text{COMBI}_{ADL}$ |
| 少 | ☆ 1 | 1.860 | 1.577 | 1.792 | 2.689 | 1.987 | 2.283 | **0.784** | 0.867 | **0.755** | 0.815 | 0.815 |
| 少 | ☆ 2 | 1.440 | 1.339 | 1.569 | 1.841 | 1.584 | 1.818 | **0.869** | 1.191 | **1.069** | 1.220 | 1.210 |
| 多 | ☆ 3 | **0.982** | **0.991** | 1.193 | 1.197 | 1.178 | 1.223 | 1.021 | 1.192 | 1.132 | 1.295 | 1.315 |
| 多 | ☆ 4 | **0.580** | 0.820 | 0.833 | **0.720** | 0.799 | 0.772 | 0.982 | 0.777 | 0.926 | 0.927 | 0.928 |
| 多 | ☆ 5 | 0.763 | 0.839 | **0.616** | **0.588** | 0.622 | 0.701 | 0.973 | 0.808 | 0.907 | 0.723 | 0.711 |
| | micro ave. | **0.806** | 0.895 | **0.869** | 0.888 | 0.867 | 0.923 | 0.979 | 0.878 | 0.952 | 0.916 | 0.916 |
| | macro ave. | 1.125 | 1.113 | 1.201 | 1.407 | 1.234 | 1.359 | **0.919** | 0.967 | **0.958** | 0.996 | 0.996 |

均はおおよそ 1.1 から 1.4 程度と，micro 平均に比べると悪くなっている．偏ったデータに適した損失関数を適用しても，抜本的な精度改善は達成できなかった．

LLM 単体の手法（LLM）の少数派の精度は良い．$\text{BERT}_{***}$ の中で最良の $\text{BERT}_{WCE}$ では，☆ 1 で 1.577，☆ 2 で 1.339 であったのに対し，LLM では 0.784 と 0.869 になった．LLM は追加の学習を必要としないため，☆ 1 と☆ 2 の学習データの少なさによる影響を受けなかったと思われる．また，低評価なレビュー文書はネガティブな記述で一貫する傾向が強いため，LLM による推定が容易であったとも考えられる．一方で多数派の精度は，$\text{BERT}_{***}$ から悪化した．これは，追加学習を実施しなかったことで，LLM が評価値ごとの細かな特徴を理解できなかったからだと考えられる．少数派の大幅な精度改善によって macro 平均が 0.919 まで改善したが，多数派の悪化によって micro 平均が 0.979 まで悪化した．

両者を組み合わせた手法（$\text{COMBI}_{***}$）では，$\text{BERT}_{***}$ が苦手とする少数派で精度が改善した．少数派の精度が改善したため，$\text{BERT}_{***}$ よりも macro 平均が改善した．また，LLM が苦手としていた多数派でも，☆ 3 以外で LLM より精度が改善した．そのため，micro 平均も LLM より改善した．LLM で少数派クラスの精度を改善しつつ多数派クラスの精度も維持でき，少数派クラスと多数派クラスの精度をバランスよく保てる点で，両者を組み合わせた提案手法は有効だといえる．

### 4.4　データ拡張アプローチとの比較

本研究では，データ数の調整が不要な手法を提案したが，データ拡張アプローチもしばしば用いられる [5][6]．そこで，提案手法をデータ拡張アプローチと比較することで，さらなる有効性を検証する．具体的には，各観点の☆ 1 と☆ 2 のデータ数を☆ 3 と同じ水準まで増加させて実験した．増加させる☆ 1 と☆ 2 のデータは，楽天トラベルデータ[2)]からランダムで選択した．

**表 2**　データ拡張手法との比較結果．各評価値で最良の手法は「**太字+下線**」で示している．

| 6 観点平均 | | $\text{BERT}_{MSE}$ | $\text{COMBI}_{MSE}$ | $\text{BERT}_{arg}$ |
|---|---|---|---|---|
| 少 | ☆ 1 | 1.860 | **0.867** | 1.198 |
| 少 | ☆ 2 | 1.440 | **1.191** | 1.243 |
| 多 | ☆ 3 | **0.982** | 1.192 | 1.146 |
| 多 | ☆ 4 | **0.580** | 0.777 | 0.834 |
| 多 | ☆ 5 | **0.763** | 0.808 | 0.785 |
| | micro ave. | **0.806** | 0.878 | 0.886 |
| | macro ave. | 1.125 | **0.967** | 1.041 |

拡張後の学習データを用いて $\text{BERT}_{MSE}$ を学習させた結果（$\text{BERT}_{arg}$）を表 2 に示す．表 2 より，$\text{BERT}_{arg}$ の精度は少数派クラスで $\text{BERT}_{MSE}$ より改善したものの，$\text{COMBI}_{MSE}$ より良くなることは無かった．このことからも提案手法の有効性がうかがえる．

## 5　おわりに

本研究では，低評価なレビュー文書は高評価なレビューに比べて少数派になるという着目点から，教師有り学習モデルと LLM を組み合わせてレビュー文書の評価値推定タスクに取り組んだ．教師有り学習モデルのみ，もしくは LLM のみによる推定と比較して，両者を統合した提案手法はバランス良く全体の評価値を推定できる手法であることを確認した．さらに，一般によく用いられるデータ拡張手法との比較からも，提案手法の有効性を確認した．

今回は少数派が低評価レビューであるという条件であったが，基本的にはその逆（高評価レビューが少数派）の場合にも提案手法は適用可能である．提案手法のさらなる有効性の確認のために，異なる分布のデータセットによる評価などが必要である．

2)　楽天グループ株式会社 (2020): 楽天トラベルデータ. 国立情報学研究所情報学研究データリポジトリ. (データセット). https://doi.org/10.32130/idr.2.2


This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

# 参考文献


[1] Bo Pang and Lillian Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In **Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)**, pp. 115–124, 2005.

[2] Daisuke Okanohara and Jun'ichi Tsujii. Assigning polarity scores to reviews using machine learning techniques. In **Second International Joint Conference on Natural Language Processing: Full Papers**, 2005.

[3] Takuto Nakamuta and Kazutaka Shimada. Multi-aspects rating prediction using aspect words and sentences. In **Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation**, pp. 513–521, 2015.

[4] Masaki Takeo, Shinnosuke Kawasaki, and Kazutaka Shimada. Rating prediction of multi-aspect reviews using simultaneous learning. In **2023 International Conference on Asian Language Processing (IALP)**, pp. 358–363, 2023.

[5] Sophie Henning, William Beluch, Alexander Fraser, and Annemarie Friedrich. A survey of methods for addressing class imbalance in deep-learning based natural language processing. In **Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics**, pp. 523–540, 2023.

[6] Daphné Chopard, Matthias S. Treder, and Irena Spasić. Learning data augmentation schedules for natural language processing. In **Proceedings of the Second Workshop on Insights from Negative Results in NLP**, pp. 89–102, 2021.

[7] 川嵜慎乃介, 嶋田和孝. アスペクト文判別のための大規模言語モデルを用いたデータ拡張の有効性. 情報処理学会九州支部 火の国情報シンポジウム, pp. B–4–4, 2024.

[8] Yinhao Bai, Zhixin Han, Yuhua Zhao, Hang Gao, Zhuowei Zhang, Xunzhi Wang, and Mengting Hu. Is compound aspect-based sentiment analysis addressed by LLMs? In **Findings of the Association for Computational Linguistics: EMNLP 2024**, pp. 7836–7861, 2024.

[9] Jakub Šmíd, Pavel Priban, and Pavel Kral. LLaMA-based models for aspect-based sentiment analysis. In **Proceedings of the 14th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis**, pp. 63–70, 2024.

[10] Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. SemEval-2015 task 12: Aspect based sentiment analysis. In **Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)**, pp. 486–495, 2015.

[11] Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Nuria Bel, Salud María Jiménez-Zafra, and Gülşen Eryiğit. SemEval-2016 task 5: Aspect based sentiment analysis. In **Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)**, pp. 19–30, 2016.

[12] Xiaoya Li, Xiaofei Sun, Yuxian Meng, Junjun Liang, Fei Wu, and Jiwei Li. Dice loss for data-imbalanced NLP tasks. In **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics**, pp. 465–476, 2020.

[13] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In **2017 IEEE International Conference on Computer Vision (ICCV)**, pp. 2999–3007, 2017.

[14] Daiki Nishiyama, Kazuto Fukuchi, Youhei Akimoto, and Jun Sakuma. Camri loss: Improving the recall of a specific class without sacrificing accuracy. **IEICE Transactions on Information and Systems**, pp. 523–537, 2023.

[15] Heike Adel, Francine Chen, and Yan-Ying Chen. Ranking convolutional recurrent neural networks for purchase stage identification on imbalanced Twitter data. In **Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers**, pp. 592–598, 2017.

[16] Cícero dos Santos, Bing Xiang, and Bowen Zhou. Classifying relations by ranking with convolutional neural networks. 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. 626–634, 2015.

[17] Yuki Nakayama, Koji Murakami, Gautam Kumar, Sudha Bhingardive, and Ikuko Hardaway. A large-scale Japanese dataset for aspect-based sentiment analysis. In **Proceedings of the Thirteenth Language Resources and Evaluation Conference**, pp. 7014–7021, 2022.

[18] 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.

[19] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In **2016 Fourth International Conference on 3D Vision (3DV)**, pp. 565–571, 2016.




This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

# A　損失関数の式

BERT モデルに適用した各損失関数は以下の式で表される．$N$ はデータ数，$i$ は $i$ 番目のデータである．また，$s$ は 1 から 5 の評価値，$y_{is}$ は，評価値 $s$ が正解か不正解かを表すフラグ，$p_{is}$ は評価値 $s$ に対する予測確率，$y_{it}$ は正解評価値 $t$ を表すフラグ，$p_{it}$ は正解評価値 $t$ に対する予測確率である．各損失関数固有の記号や，ハイパーパラメータは各節で説明する．

## A.1　MSE

$$MSE = \sum_{i=1}^{N} (\hat{y}_i - y_i)^2 \tag{1}$$

$\hat{y}_i$ はモデルの推定回帰値，$y_i$ は正解回帰値である．

## A.2　WCE

$$WCE = -\frac{1}{N} \sum_{i=1}^{N} \alpha_i \sum_{s=1}^{5} y_{is} log(p_{is}) \tag{2}$$

重み係数 $\alpha_i$ は，[12] に倣い式 3 で定義する．ここで，$\#_{all}$ は全データの数，$\#_{is}$ は $i$ 番目のデータが属する評価値 $s$ のデータ数である．$K$ はハイパーパラメータである．本研究では 1 とする．

$$\alpha_i = \log_{10}(\frac{\#_{all} - \#_{is}}{\#_{is}} + K) \tag{3}$$

## A.3　FL

$$FL = -\frac{1}{N} \sum_{i=1}^{N} (1 - p_{it})^k \sum_{s=1}^{5} y_{is} log(p_{is}) \tag{4}$$

$k$ はハイパーパラメータである．$\text{BERT}_{FL}$ で 3，$\text{COMBI}_{FL}$ で 2 とする．

## A.4　DL

$$DL = 1 - \frac{2\sum_i^N p_{it} y_{it} + \Upsilon}{\sum_i^N {p_{it}}^2 + \sum_i^N {y_{it}}^2 + \Upsilon} \tag{5}$$

$\Upsilon$ はハイパーパラメータである．本研究では 1 とする．

## A.5　ADL

$$ADL = 1 - \frac{2\sum_i^N (1 - p_{it}) p_{it} y_{it} + \Upsilon}{\sum_i^N (1 - p_{it}) p_{it} + \sum_i^N y_{it} + \Upsilon} \tag{6}$$

$\Upsilon$ はハイパーパラメータである．本研究では 1 とする．

## A.6　RL

$$RL = -\frac{1}{N} \sum_{i=1}^{N} ( log\left(1 + e^{k(m^+ - p_{it})}\right) + log\left(1 + e^{k(m^- + c^-)}\right)) \tag{7}$$

$c^-$ は，正解評価値 $t$ 以外で最も予測確率が高いクラス $c$ の予測確率である．$k$，$m^+$，$m^-$ はそれぞれハイパーパラメータである．本研究ではそれぞれ 2，0.5，0.2 とする．

# B　BERT モデルのハイパーパラメータ

BERT モデルには，東北大学が公開しているモデル[3)]を使用した．エポック数は 5，バッチサイズは 4，最大トークン長は 512 に設定した．学習率は $\text{BERT}_{RL}$ で 5e-6，それ以外の $\text{BERT}_{***}$ で 1e-6，$\text{COMBI}_{***}$ で 5e-7 とした．

# C　LLM へのプロンプト

GPT-4o-mini への入力プロンプトを図 3 に示す．**{aspect}**には評価値を推定する観点名が，**{review}**にはレビュー文書が，それぞれ当てはめられる．

System Role
You are a good assistant to predict the rating score of reviews written in Japanese.

User Prompt
次のレビュー文書内の「**{aspect}**」に対する記述を踏まえて、「**{aspect}**」に対する感情度合いを「1、2、3、4、5」の整数5段階で出力してください。1に近づくほど否定的、5に近づくほど肯定的になります。もし「**{aspect}**」に対する記述がない場合は，他の観点への記述をもとに推測してください。
『**{review}**』。
出力は，次のJsonフォーマットの通り出力してください。
{"star":感情度合い（1〜5）,"evidence":その理由}

**図 3**　GPT-4o-mini への入力プロンプト．

3)　https://huggingface.co/tohoku-nlp/bert-base-japanese-whole-word-masking


This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).