文分類問題における精度と解釈性向上のための近傍事例の活用

村岡 雅康[1]　趙 陽[2]
日本アイ・ビー・エム株式会社 東京基礎研究所
[1]mmuraoka@jp.ibm.com　[2]yangzhao@ibm.com

概要

プロンプトを用いた zero/few-shot 評価は，ファインチューニングを行うことなく大規模言語モデルの下流タスクにおける性能評価を可能にする一方で，性能面に改善の余地がある．本稿では，大規模言語モデルからの特徴量を用いて k 近傍法を行うことで，zero/few-shot 評価の枠組みにおいて性能向上を達成する手法を提案する．また，k 近傍事例を参照することで，モデルの予測結果に対する解釈性の向上も期待される．評価実験により，極性分類，トピック分類，含意関係認識を含む文分類問題（6 つのデータセット）において，近傍事例を活用することで提案手法の性能が大幅に向上することを示す．

1　はじめに

BERT [1] や GPT-2 [2]，T5 [3] のように大量のテキストデータで事前学習された大規模言語モデルが，様々な下流タスクに適用され成功を収めている．これらの言語モデルの性能評価を様々なタスクおよびドメインで行うことは実応用において重要である．

評価方法の一つに下流タスクでファインチューニングを行わない zero-shot/few-shot 評価がある [4]．これらは，下流タスクを言語モデルタスクに変換して評価を行う方法である．この評価方法は，ファインチューニングの学習アルゴリズムや，そのハイパーパラメタ，モデルパラメタの更新などファインチューニング由来の要因を排除して言語モデルの性能評価を行うことができるという特長がある．

一方で，これらの評価方法において，使用するプロンプトに評価結果が大きく左右されたり [5, 6]，言語モデルの予測にバイアスが乗るといった問題が報告されている [7]．Zhao らはこれらの問題の調査・分析を行い，バイアスを除去することで，複数の下流タスクにおいて言語モデルの性能向上を達成した [7]．しかしながら，性能面においてはまだ改善の余地が残されており，また，予測結果からモデルの予測理由を推察することが困難である．

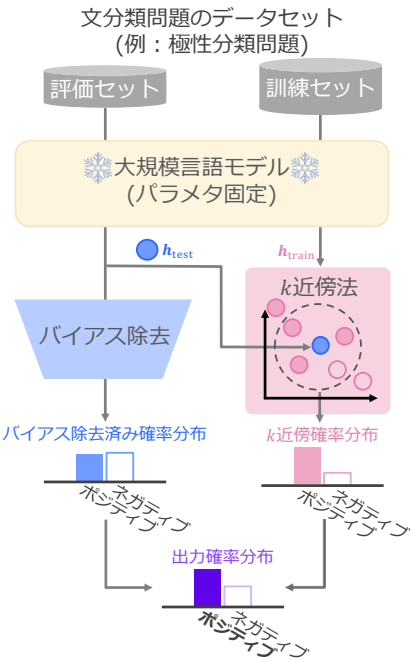

図 1: 提案手法の概要図

そこで本稿では，評価対象である大規模言語モデルから抽出した特徴量を用いて k 近傍法 [8, 9] を行うことで，下流タスクでの更なる性能向上を目指す．具体的には，図 1 に示すように，バイアス除去を行う既存手法の予測確率と，近傍事例からの予測確率を線形補完することで最終的な予測結果を得る．また，本手法は近傍事例を参照することで，モデルが入力文をどう解釈したかを推測することができるという点で，解釈性の向上にも寄与する．

本手法を多様なドメインのテキスト，および，極性分類 [10]，トピック分類 [11, 12]，含意関係認識 [13, 14] を含む複数の文分類問題に適用し評価実験を行ったところ，近傍事例を活用することで性能が大幅に向上すること確認した．また，定性分析により，たとえ提案手法が予測を誤った場合でも，近傍事例からその理由を推測できることもわかった．

2 関連研究

プロンプトを用いた評価は，プロンプトと呼ばれるタスクを解くための指示テキストを用いて，下流タスクを言語モデルタスクに変換し評価を行う方法である．特に，下流タスクで一切ファインチューニングを行わない方法を Tuning-free と呼ぶ [15]．本研究における提案手法も Tuning-free な評価方法である．この評価方法では，zero-shot 評価と in-context few-shot 評価が可能である．zero-shot 評価は，言語モデルに下流タスクの評価セットの事例のみを提示し答えさせる方法である．in-context few-shot 評価は，評価事例の前にコンテキストとして訓練セットから抽出した一つ以上の事例を正解付きで追加した状態でモデルに提示し，予測させる方法である．

Tuning-free の評価方法において，前節で述べた Zhao らの研究 [7] のほかにいくつか関連研究が存在する．Holtzman ら [16] は評価タスクのドメイン情報を自己相互情報量 (PMI) によって割り引いた予測確率を求める手法を提案している．Kassner と Schütze [17] は情報検索器を言語モデルに付け加えて k 近傍法を行う手法を提案している．この手法は新たなモデルパラメタを増やすが，我々の手法は評価対象の言語モデルを特徴量抽出器としてそのまま k 近傍法に使用するため，新たなモデルパラメタを増やさずに済む．Shi ら [18] は言語モデルに k 近傍言語モデル [9] を使用し，zero-shot 評価での性能向上を達成した．本稿では，Zhao らの手法で得られるバイアスが除去された予測確率と，k 近傍言語モデルで使用されているスコア関数を用いて得られる k 近傍予測確率を組み合わせて更なる性能向上を目指す．特に，Kassner と Schütze や Shi らは下流タスクの訓練セットに含まれる正解ラベル情報を使わなかったが，本研究では明示的にこれを使用する．そのほかのプロンプトを用いた評価の関連研究については，文献 [15] のサーベイ論文を参照されたい．

3 問題設定

本研究では，文分類問題における大規模言語モデルのプロンプトを用いた zero/few-shot 評価を行う．大規模言語モデルは事前学習済みのものを使用し，評価中にはモデルパラメタの更新，すなわち，ファインチューニングは行わない．few-shot 評価においては，in-context few-shot のように予測の際に言語モデルに少量の訓練事例を提示するのではなく，k 近傍法によって取得される少量の訓練事例を予測の計算に使用する．我々はこれを「example-based few-shot 評価」と呼ぶ．

極性分類問題を例として，基本的なプロンプトを用いた zero/few-shot 評価について説明する．[1] 文分類問題は，入力文 x が与えられた時に，出力ラベル $y \in \mathcal{Y}$ を予測する多クラス問題である．$\mathcal{Y}$ はラベル集合であり，事前に定義されているものとする．例えば，映画のレビュー文に対する極性分類問題 [10] では，$\mathcal{Y} = \{$ ポジティブ, ネガティブ $\}$ などとなる．データセット $\mathcal{T}$ は，入力文と正解の出力ラベルのペア (x, y^*) の集合からなり，評価セット $\mathcal{T}_{\text{test}}$ と訓練セット $\mathcal{T}_{\text{train}}$ に分かれているものとする．[2]

この問題を解くためには言語モデルを用いて出力ラベルに関する条件付き確率分布 $p(y|x)$ を求める必要があるが，モデルのファインチューニングは行わないため，このままでは上記の条件付き確率分布を直接モデル化することができない．そこで，これを穴埋め形式の言語モデルタスクとして定式化する．

事前学習済み大規模言語モデル M は，入力文中の `[MASK]` トークンの位置に相応しい単語を予測する：

$$p(w_l | \boldsymbol{w}_{\backslash l}; M). \tag{1}$$

ただし，$\boldsymbol{w}_{\backslash l}$ は，入力文 $x = w_1 \ldots w_L$ のうち位置 l $(1 \leq l \leq L)$ の単語 w_l が `[MASK]` トークンに置き換えられた文である．ここで，極性分類問題を解かせるために，プロンプト f_{prompt} を用いて言語モデルに解き方をテキストで指示する：

$$f_{\text{prompt}}(x, L) = \underline{\text{Review:}}\ [x]\ \underline{\text{Sentiment:}}\ \texttt{[MASK]} \tag{2}$$

$[x]$ には任意の入力文が挿入される．プロンプト適用済み入力文を $x'_{\backslash L} = f_{\text{prompt}}(x, L)$ と表し，言語モデルに入力することで，`[MASK]` トークンの位置に入る単語の予測確率分布 $p(w_L | x'_{\backslash L}; M)$ を得る．これは言語モデルが扱う全ての語彙 $\mathcal{V}$ 上の確率分布であることに注意されたい．ここから極性分類の出力ラベル集合 $\mathcal{Y}$ の各要素に対応する確率値のみ抽出し，合計が 1 になるように正規化すれば，所望の確率分布 $p(y|x; M)$ が得られる．[3]

4 提案手法

提案手法は，前節のプロンプトを用いた zero/few-shot 評価方法に 2 点改良を加えたものである．

1) これでも文分類問題の定式化としての一般性は失われない．
2) 開発セットも含めた 3 つに分かれている場合もある．
3) 従って，$\mathcal{Y} \subset \mathcal{V}$ を満たさなければならない．満たしていない場合は，適宜ラベル集合 $\mathcal{Y}$ を再定義する必要がある．

(1) 言語モデルの予測バイアスの除去 [7][4)]

(2) 言語モデルの特徴量を用いた k 近傍法 [8, 9]

図 1 に概要を示した通り，提案手法の予測する出力ラベルの確率分布は上記 2 つの仕組みからそれぞれ得られる確率分布の線形和である．

$$\hat{p}(y|x; M, \mathcal{D}) = \lambda \hat{p}_{\text{debias}} + (1-\lambda)\hat{p}_{\text{kNN}} \tag{3}$$

$\mathcal{D}$ は k 近傍法で使用するデータストアであり，$\lambda \in [0, 1]$ は線形補完係数である．

バイアス除去 [7] は言語モデルの予測に乗ったバイアスを，解きたい文分類問題に関係のない文字列を使って除去する方法である．“N/A” や空文字列などそれ単体では意味を持たないような文字列を，極性分類問題の入力文だと仮定し，言語モデルに解かせた場合を考える．それらの文字列に極性はないと考えるのが自然なので，ポジティブ，ネガティブともに 50% と予測されるべきだが，実際はポジティブ = 63%，ネガティブ = 37% のようにバイアスが乗った予測になることが報告されている [7].

これらのバイアスをなくすように確率分布を補正するには，上記のような文分類問題に関係のない文字列を使って得られた確率値の逆数を補正項として乗ずることで実現できる．本研究では Zhao らの研究 [7] で用いられた，$\mathcal{C}$ = {“N/A”, “”, “[MASK]”} の 3 種類を使用する．補正項 $\hat{p}_{\text{cf}}$ は以下で求められる[5)]：

$$\hat{p}_{\text{cf}} = \frac{1}{\mathcal{C}} \Sigma_{c \in \mathcal{C}} \hat{p}(y|c'_{\backslash L}; M). \tag{4}$$

ただし，$c'_{\backslash L} = f_{\text{prompt}}(c, L)$ である．バイアス除去を用いた出力ラベルの確率分布は次のようになる：

$$\hat{p}_{\text{debias}} = W_{\text{debias}} \hat{p}(y|x'_{\backslash L}; M), \tag{5}$$

$$W_{\text{debias}} = \text{diag}(\hat{p}_{\text{cf}})^{-1}. \tag{6}$$

$\text{diag}(\hat{p}_{\text{cf}})$ は $\hat{p}_{\text{cf}}$ から対角行列を作成する関数である．

k 近傍法は，評価セットの入力文 x を言語モデルでエンコードした特徴量 $\boldsymbol{h}_{\text{test}}$ をクエリとして，データストアに対して k 近傍探索を行い，得られた近傍事例を入力文 x に類似した事例とみなし，それらの正解ラベル y^*_{train} を予測に使用する方法である．近傍事例から予測を求める方法は複数存在するが [19, 20]，本研究では，近年提案された k 近傍言語モデル [9] で用いられている方法を採用する．

4) バイアス除去を行わず元の確率分布をそのまま使用することも可能である．

5) cf は context-free の頭文字である．

表 1: 文分類問題のデータセット

データセット	タスク	ドメイン	クラス数	訓練事例数	評価事例数
SST-2 [10]	極性分類	レビュー	2	6,920	1,821
DBPedia [11]	トピック分類	Wikipedia	14	49,999	70,000
AGNews [11]	トピック分類	ニュース	4	120,000	7,600
TREC [12]	質問分類	質問文	6	5,452	500
QNLI [13]	含意関係/QA	Wikipedia	2	104,743	5,463
MNLI [14]	含意関係	多ドメイン	3	150,000	9,815

事前に訓練セット $\mathcal{T}_{\text{train}}$ から k 近傍法のためのデータストア $\mathcal{D}$ を構築しておく．

$$\mathcal{D} = \{(\boldsymbol{h}_{i,L}, x_i, y_i) | (x_i, y_i) \in \mathcal{T}_{\text{train}}\} \tag{7}$$

$\boldsymbol{h}_{i,L} \in \mathbb{R}^d$ は言語モデル M で入力文 $x'_{i,\backslash L}$ をエンコードした時の位置 L の最終隠れ層の特徴量ベクトル[6)]である．同様に評価セット内の入力文 $x'_{\backslash L}$ に対しても言語モデルから特徴量 $\boldsymbol{h}_{\text{test}}$ を得る．続いて k 近傍探索によってデータストア $\mathcal{D}$ から特徴量 $\boldsymbol{h}_{\text{test}}$ に最も近い k 個の近傍事例 (の集合) $\mathcal{N}$ を得る．

$$\mathcal{N} = \text{argmin-k}_{(\boldsymbol{h}_i, x_i, y_i) \in \mathcal{D}} d(\boldsymbol{h}_{\text{test}}, \boldsymbol{h}_i) \tag{8}$$

$d(\cdot, \cdot)$ は距離関数であり，k 近傍言語モデル [9] に倣い，L2 距離を使用する．argmin-k は上位 k 個の事例を返すように拡張された argmin 関数である．最終的な k 近傍事例を用いた確率分布を次式で求める：

$$\hat{p}_{\text{kNN}} \propto \frac{1}{|\mathcal{N}|} \Sigma_{(\boldsymbol{h}_i, \_, y_i) \in \mathcal{N}} \mathbb{1}_{\mathcal{Y}}(y_i) \exp\left(-d(\boldsymbol{h}_i, \boldsymbol{h}_{\text{test}})/T\right) \tag{9}$$

$\mathbb{1}_{\mathcal{Y}}(y_i)$ は y_i の次元のみ 1，それ以外は 0 の one-hot ベクトル（次元数：$|\mathcal{Y}|$）を返す関数であり，$\exp(\cdot)$ の項は k 近傍言語モデル [9] で使用されている距離関数に基づく確率分布である（T は温度パラメータで実験では $T = 1000$ を使用[7)]）．実際の $\hat{p}_{\text{kNN}}$ は和が 1 になるように正規化を行い使用する．

5 実験

文分類問題でよく使われる 6 つのデータセットを用いて評価実験を行う．表 1 に概要を示す．詳細は付録 A に記載した．事前学習済み大規模言語モデルとして RoBERTa [21] を使用した．[8)] バイアス除去を用いない手法 (No debias) [4]，バイアス除去手法 [7] (Debias)，および，k 近傍法のみを用いる手法 (kNN only) を比較手法とした．[9)] 前者 2 手法は訓練セットから無作為に抽出された k 個の事例を用いて

6) 特に断りがない場合，$d = 768$ とし，$\boldsymbol{h}_{i,L}$ を $\boldsymbol{h}_i$ と略記する．

7) 予備実験より，異なる値でもほとんど同じ結果が得られた．

8) https://huggingface.co/roberta-base [22].

9) 色は図 1 および式 (3) 内の色と対応し，No Debias は W_{debias} がない式 (5) に対応する．

表 2: 文分類問題の実験結果 (精度). **太字**は列内での，下線はデータセット内でのそれぞれ最大精度を表し，下付き数字は標準偏差 (± 記号あり) または λ の値 (± 記号なし) を表す．

(a) SST-2

手法	0-shot	1-shot	4-shot	8-shot
No Debias	63.5	$62.9_{\pm 2.7}$	$62.0_{\pm 12.4}$	$50.9_{\pm 1.8}$
+kNN	63.5	$85.2_{0.6}$	$85.8_{0.4}$	$87.5_{0.2}$
Debias	**85.6**	$76.8_{\pm 3.3}$	$78.1_{\pm 5.8}$	$73.3_{\pm 7.8}$
+kNN	**85.6**	$\mathbf{86.6}_{0.9}$	$\mathbf{88.3}_{0.6}$	$\underline{\mathbf{89.3}}_{0.4}$
kNN only	N/A	$82.9_{0.0}$	$85.6_{0.0}$	$87.9_{0.0}$

(b) DBPedia

手法	0-shot	1-shot	4-shot	8-shot
No Debias	18.0	$14.7_{\pm 9.6}$	$25.2_{\pm 13.1}$	$21.8_{\pm 12.7}$
+kNN	18.0	$\mathbf{85.4}_{0.0}$	$88.0_{0.0}$	$\underline{\mathbf{88.4}}_{0.2}$
Debias	**31.1**	$56.6_{\pm 5.3}$	$53.6_{\pm 8.1}$	$47.9_{\pm 11.0}$
+kNN	**31.1**	$\mathbf{85.4}_{0.6}$	$\mathbf{88.1}_{0.4}$	$\underline{\mathbf{88.4}}_{0.3}$
kNN only	N/A	$\mathbf{85.4}_{0.0}$	$87.9_{0.0}$	$88.3_{0.0}$

(c) AGNews

手法	0-shot	1-shot	4-shot	8-shot
No Debias	**69.7**	$47.9_{\pm 11.6}$	$64.1_{\pm 9.4}$	$47.0_{\pm 11.9}$
+kNN	**69.7**	$\mathbf{89.0}_{0.0}$	$\mathbf{91.3}_{0.3}$	$91.4_{0.3}$
Debias	65.8	$67.9_{\pm 5.2}$	$68.4_{\pm 8.2}$	$54.4_{\pm 13.8}$
+kNN	65.8	$\mathbf{89.0}_{0.0}$	$\mathbf{91.3}_{0.3}$	$\underline{\mathbf{91.5}}_{0.3}$
kNN only	N/A	$\mathbf{89.0}_{0.0}$	$90.9_{0.0}$	$\underline{\mathbf{91.5}}_{0.0}$

(d) TREC

手法	0-shot	1-shot	4-shot	8-shot
No Debias	**48.4**	$37.8_{\pm 5.2}$	$24.6_{\pm 2.8}$	$20.7_{\pm 4.2}$
+kNN	**48.4**	$\mathbf{81.8}_{0.0}$	$\mathbf{85.4}_{0.0}$	$\underline{\mathbf{87.6}}_{0.0}$
Debias	32.0	$31.2_{\pm 3.2}$	$30.5_{\pm 5.3}$	$32.2_{\pm 6.0}$
+kNN	32.0	$\mathbf{81.8}_{0.0}$	$\mathbf{85.4}_{0.0}$	$\underline{\mathbf{87.6}}_{0.0}$
kNN only	N/A	$\mathbf{81.8}_{0.0}$	$\mathbf{85.4}_{0.0}$	$\underline{\mathbf{87.6}}_{0.0}$

(e) QNLI

手法	0-shot	1-shot	4-shot	8-shot
No Debias	**50.7**	$50.2_{\pm 0.6}$	$50.3_{\pm 1.5}$	$50.9_{\pm 0.6}$
+kNN	**50.7**	$50.7_{1.0}$	$50.7_{1.0}$	$54.5_{0.5}$
Debias	49.6	$49.6_{\pm 0.2}$	$49.2_{\pm 0.5}$	$50.1_{\pm 0.3}$
+kNN	49.6	$49.6_{1.0}$	$49.6_{1.0}$	$55.1_{0.6}$
kNN only	N/A	$\mathbf{56.8}_{0.0}$	$\mathbf{58.5}_{\mathbf{0.0}}$	$\underline{\mathbf{60.5}}_{0.0}$

(f) MNLI

手法	0-shot	1-shot	4-shot	8-shot
No Debias	**35.5**	$35.6_{\pm 2.5}$	$31.5_{\pm 1.2}$	$33.5_{\pm 1.9}$
+kNN	**35.5**	$35.5_{1.0}$	$35.5_{1.0}$	$36.3_{0.6}$
Debias	31.3	$33.9_{\pm 1.2}$	$31.6_{\pm 0.8}$	$32.7_{\pm 1.8}$
+kNN	31.3	$31.3_{1.0}$	$31.3_{1.0}$	$31.3_{1.0}$
kNN only	N/A	$\mathbf{36.8}_{0.0}$	$\mathbf{37.7}_{0.0}$	$\underline{\mathbf{39.0}}_{0.0}$

in-context few-shot 評価を行う．一方，提案手法 (* + kNN) および kNN only は，kNN によって抽出された k 個の訓練事例を用いて example-base few-shot 評価を行う．$k \in \{1, 4, 8\}$ を使用し，式 (3) における補完係数 λ は訓練セットを用いて調整した．[10)]

結果を表 2 に示す．[11)]QNLI，MNLI 以外のデータセットにおいて，バイアス除去の有無に関わらず，k 近傍法を組み合わせることで (+ kNN)，組み合わせない手法に比べ，精度が大幅に向上することを確認した．QNLI，MNLI においては，kNN only が最も良かった．これらの結果から，ファインチューニングを行わなくとも言語モデルは下流タスクを解くために有用な情報を，k 近傍法において効果的にそれが発揮される形で特徴量に埋め込むことができていると言える．[12)]また，全てのデータセットにおいて，近傍事例数 k を大きくするほど，精度が向上した．ただし，各データセットで最大精度を達成する k 近傍法と相性の良い手法の組み合わせはデータセットに依存することがわかる．このことから，任意のデータセットに対して頑健な k 近傍法と既存手法との組み合わせは本実験で用いた中にはないと言える．この頑健性の調査は今後の課題とする．

続いて，SST-2（極性分類問題）の訓練セットを用いて，提案手法の予測結果の解釈性に関する定性分析を行った．表 3 に提案手法 (**Debias + kNN**) を含む 3 手法の予測結果と提案手法で予測に使われた

表 3: SST-2 における 3 手法の予測結果

入力文 : is it a comedy? **正解ラベル** : ネガティブ

正否	手法	予測ラベル（予測確率）
×	No Debias	ポジティブ（87.85%）
×	Debias	ポジティブ（60.90%）
×	**Debias + kNN**	ポジティブ（76.54%）

Debias + kNN によって得られた k 近傍事例 ($k = 4$)

(it 's funny., ポジティブ), (a funny film., ポジティブ)
(a surprisingly funny movie., ポジティブ)
(the film grows on you., ポジティブ)

近傍事例を示す．3 手法全てが誤った予測をしているが，比較手法は予測結果しかなく，誤った予測になった原因を推測することが難しい．一方，提案手法はその原因を近傍事例から推測できる．表 3 の事例の場合，入力文に対する正解ラベルはネガティブであるにも関わらず，言語モデルからの特徴量を用いて得られた近傍事例が全てポジティブな文になっていることから，言語モデルは入力文をポジティブな文と解釈していることがわかる．[13)]このようにして，提案手法は，評価対象である言語モデルの特徴量を用いた k 近傍法の近傍事例により，言語モデルが入力文をどのように解釈しているかを推測する手がかりを提示できると考える．

6 おわりに

本稿では，プロンプトを用いた zero/few-shot 評価において，k 近傍法を使用することで精度と解釈性を向上する手法を提案した．実験結果から，複数の文分類問題において有効性を確認した．今後は，他のタスクでの有効性・頑健性を確認したい．

10) 0 から 1 まで 0.1 刻みの 11 個の λ の値のうち，訓練セットで最高精度を達成した λ の値で評価を行った．

11) in-context few-shot 評価を行った手法に関しては，使用する訓練事例によって精度にばらつきが生じるため，5 つの乱数シードを用いて得られた精度の平均値と標準偏差を報告する．

12) k 近傍法ではファインチューニングを行っていない言語モデルの特徴量を使用していることを思い出されたい．

13) 正解ラベルがネガティブなのは，入力文が疑問文であることより，映画のクオリティにネガティブな印象を持つレビュー文だからである．

参考文献

[1] 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.

[2] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[3] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, Vol. 21, No. 140, pp. 1–67, 2020.

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, Vol. 33, pp. 1877–1901, 2020.

[5] Boxi Cao, Hongyu Lin, Xianpei Han, Fangchao Liu, and Le Sun. Can prompt probe pretrained language models? understanding the invisible risks from a causal view. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5796–5808, 2022.

[6] Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. Measuring and Improving Consistency in Pretrained Language Models. *Transactions of the Association for Computational Linguistics*, Vol. 9, pp. 1012–1031, 12 2021.

[7] Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning*, pp. 12697–12706, 2021.

[8] Pádraig Cunningham and Sarah Jane Delany. K-nearest neighbour classifiers - a tutorial. *ACM Computing Surveys*, Vol. 54, No. 6, 2021.

[9] Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Generalization through memorization: Nearest neighbor language models. In *International Conference on Learning Representations*, 2020.

[10] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1631–1642, 2013.

[11] Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, Vol. 28, 2015.

[12] Ellen M. Voorhees and Dawn M. Tice. Building a question answering test collection. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '00, p. 200–207, 2000.

[13] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2383–2392, 2016.

[14] Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 1112–1122, 2018.

[15] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 2022. Just Accepted.

[16] Ari Holtzman, Peter West, Vered Shwartz, Yejin Choi, and Luke Zettlemoyer. Surface form competition: Why the highest probability answer isn' t always right. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 7038–7051, 2021.

[17] Nora Kassner and Hinrich Schütze. BERT-kNN: Adding a kNN search component to pretrained language models for better QA. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 3424–3430, 2020.

[18] Weijia Shi, Julian Michael, Suchin Gururangan, and Luke Zettlemoyer. Nearest neighbor zero-shot inference. *arXiv preprint arXiv:2205.13792*, 2022.

[19] Jianguo Zhang, Kazuma Hashimoto, Wenhao Liu, Chien-Sheng Wu, Yao Wan, Philip Yu, Richard Socher, and Caiming Xiong. Discriminative nearest neighbor few-shot intent detection by transferring natural language inference. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 5064–5082, 2020.

[20] Yan Wang, Wei-Lun Chao, Kilian Q Weinberger, and Laurens van der Maaten. Simpleshot: Revisiting nearest-neighbor classification for few-shot learning. *arXiv preprint arXiv:1911.04623*, 2019.

[21] 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 arXiv:1907.11692*, 2019.

[22] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pp. 38–45, 2020.

[23] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *7th International Conference on Learning Representations, ICLR 2019*, 2019.

[24] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic web*, Vol. 6, No. 2, pp. 167–195, 2015.

A　評価タスクおよびデータセット

ここでは，評価実験に使用したタスクおよびデータセットについて補足する．

SST-2 [10] は，極性分類問題のデータセットである．実際の映画レビューサイト rottentomatoes.com に書き込まれたレビュー文を使用している．各事例は，1 レビュー文を入力文とし，クラウドソーシングのアノテーションによって付与された極性ラベルを出力ラベルとしている．元のデータセットでは，極性ラベルの取りうる値は 5 値だが，それらを 2 値に変換して使うこともある [23].

DBPedia [11] は，トピック分類問題のデータセットである．DBPedia [24] は Wikipedia から構造化されたデータを抽出するプロジェクトであり，作成されたデータセットはデータベースおよびオントロジーの形式をとる．Zhang ら [11] は，テキスト分類問題用に，このデータセットから 14 のクラスに属するエントリのタイトルと概要文を無作為に抽出し，新たなデータセットを作成した．各事例は，概要文を入力文とし，オントロジー上でのクラスを出力ラベルとする．[14)]

AGNews [11] も，トピック分類問題のデータセットである．Gulli によって作成されたニュースコーパス[15)]を元にしており，Zhang ら [11] は，このコーパスの中で記事数が最も大きい 4 つのクラスに属する記事を無作為に抽出し，トピック分類用のデータセットとした．各事例は，ニュース記事本文とその記事が属するトピックのペアからなる．[16)]

TREC [12] は，質問文に関する分類問題である．異なる複数のニュースサイトから集められた大量の文書に対する質問文があり，人手で適切な質問文のみ選ばれている．また，各質問文には何に関する質問かを示すカテゴリが付与されている (全 6 種類)．TREC データセットの各事例は，質問文を入力文とし，カテゴリを出力ラベルとする．[17)]

QNLI [13] は，質問応答 (QA) に関する含意関係認識問題である．SQuAD [13] と呼ばれる質問応答データセットから Wang らが文分類問題に変換した [23]．具体的には，Wikipedia から収集したパラグラフ中から，質問文の答えを探すタスクだったものを，質問文の答えがパラグラフ中に含まれているかを 2 値で答える問題に変換した．したがって，各事例は，パラグラフと質問のペアが入力，回答がパラグラフ中に含まれているかどうかの 2 値が出力となる．プロンプトによってパラグラフと質問文が結合され，1 文としてモデルに与えられる．

MNLI [14] は，含意関係認識問題である．話し言葉や書き言葉を含む 10 個の異なるドメインから収集したテキストペアに対して，一方が他方を含意しているかをクラウドソーシングによってアノテーションしている．また，ドメイン外の性能評価を行うため，5 つのドメインに関しては，訓練セットに全く含まれていない．本研究では，訓練セットに含まれている 5 つのドメインから各 30,000 事例のみ無作為に抽出し，データストアを構築した．また，評価は訓練セットに含まれているドメインの評価セット (MNLI-matched) を使用した．各事例のデータ形式は，QNLI と同じである．[18)]

14）14 個のクラスは次の通り：Company, EducationalInstitution, Artist, Athlete, OfficeHolder, MeanOfTransportation, Building, NaturalPlace, Village, Animal, Plant, Album, Film, WrittenWork.

15）http://groups.di.unipi.it/~gulli/AG_corpus_of_news_articles.html

16）4 つのクラス（トピック）は次の通り：World, Sports, Business, Science/Tech.

17）6 つのカテゴリは次の通り：Number, Location, Person, Description, Entity, Abbreviation.

18）10 個のドメインは次の通り．訓練セットと同一ドメイン：Fiction, Government, Slate (Magazine), Telephone, Travel. 訓練セットのドメイン外：9/11, Face-to-face, Letters, OUP (Oxford University Press), Verbatim.