令和 2 年度茨城大学大学院理工学研究科情報工学専攻

修士学位論文

BERT を利用した Zero-shot 学習による同音異義語の誤り検出

所属　情報工学専攻

著者　藤井真 (19NM727R)

指導教員　新納浩幸 教授

令和 3 年 2 月 5 日 (金)

令和 2 年度茨城大学大学院理工学研究科情報工学専攻

修士学位論文

BERT を利用した Zero-shot 学習による 同音異義語の誤り検出

著者: 藤井真 (19NM727R)

指導教員: 新納浩幸 教授

論文要旨

近年，社会環境の変化を背景として電子的に日本語文を入力する機会は増加している．政府の学習指導要領には情報活用能力が記載され，小学校の段階で文字入力などの基本的な情報端末の操作を習得することが挙げられた．また，令和初頭の現在は疫学的な視点から実効性あるリモートワークが要請されている．PC やスマートフォン，タブレット端末などの情報端末は利用者層を低年齢化して普及する傾向にある．

このような利用環境下で日本語の音だけを頼りに平仮名入力し，十分な識別認識を欠いたまま変換後の漢字を用いるといった状況も散見される．特に「追求」と「追及」のような，意味や用法の差異が小さい同音異義語について顕著である．そのため，これら誤りを自動検出し再確認を促すシステムは教育的な視点からも重要性を高めている．

本論文では日本語文中に生じる同音異義語の誤りを検出する手法を提案する．言語モデル BERT を中心とした事前学習モデルにより，大規模コーパスによって事前学習した言語モデルを様々な言語タスクに適用する有効性が示された．本手法では，この BERT の持つ言語空間を用いて同音異義語の誤り検出を行う．BERT の一般的な利用形式は各言語タスクに合わせて，関連するデータにより再学習させる形式である．ここでは Zero-shot 学習の形式をとりいれることで，再学習を経ずに検出する手法を提案する．これにより機械学習の学習コストを抑えつつ，同音異義語の誤り検出について良好な結果が得られることを期待する．

同音異義語については，同音類義語と呼ばれるような意味や用法の差異が小さい語を対象とした．実験のデータには毎日新聞記事から対象の同音異義語が含まれる文を抜き出して用いる．

実験の結果から，人が細かい意味や用法の差異を頼りに識別する同音異義語の誤りについて検出できることが示された．一方で，人が識別しやすい同音異義語について大きく精度を崩す結果も見られた．これらの実験結果から事前学習モデルの一部に過剰適合が生じている推察が得られた．

Master's Thesis in Scholastic 2020,
Major in Computer and Information Sciences,
Granduate School of Science and Engineering, Ibaraki University

Detection of Homophone Errors by Zero-shot Learning using BERT

Auther: Shin Fujii (19NM727R)

Adviser: Prof. Hiroyuki Shinnou

Abstract

In recent years, the situation of electronically inputting Japanese sentences has increased in the background of changes in the social environment. The goverment curriculum guidelines described the ability to utilize information, and it mentioned that students should learn basic information terminal operations such as character input at the elementary school stage. In addition, as of early 2020s, effective remote work is required from the epidemiological point of view. Information terminals such as PCs, smartphones, and tablet terminals tend to spread to younger age groups.

In such a usage environment, there are some situations where hiragana is input by relying only on Japanese sounds and the converted kanji is used without sufficient identification. This is especially noticeable for homophones such as Japanese "tuikyu" which means "pursuit" and Japanese "tuikyuu" which means "question severely", which have small differences in meaning and usage. Therefore, a system that automatically detects these errors and encourages reconfirmation is becoming more important from an educational point of view.

In this paper, we propose a method for detecting errors in homophones that occur in Japanese sentences. The language model BERT of the pre-learning model showed the effectiveness of applying the language model pre-learned by a large corpus to various language tasks. In our method, error detection of homophones is performed using the language space of this BERT. The general usage format of BERT is to relearn with related data according to each language task. We propose a method for detecting without re-learning by adopting the Zero-shot learning format. It is expected that this will give good results for error detection of homophones and suppress the learning cost of machine learning.

We targeted homophones with only slight differences in meaning and usage. We used sentences containing homophones extracted from Mainichi newspaper articles as experimental data.

The experimental results showed that our proposed method can identify homophones with slight differences in meaning and usage. On the other hand, some had poor results for homophones that humans can easily identify. From these experimental results, it was inferred that overfitting occurred in a part of the pre-learning model.

目次

表目次

図目次

第 1 章

序論

1.1 研究概要

近年，社会環境の変化を背景として電子的に日本語文を入力する機会は増加している．政府の学習指導要領には情報活用能力が記載され，小学校の段階で文字入力などの基本的な情報端末の操作を習得することが挙げられた．また，令和初頭の現在は疫学的な視点から実効性あるリモートワークが要請されている．PC やスマートフォン，タブレット端末などの情報端末は利用者層を低年齢化して普及する傾向にある．

このような利用環境下で日本語の音だけを頼りに平仮名入力し，十分な識別認識を欠いたまま変換後の漢字を用いるといった状況も散見される．特に「追求」と「追及」のような，意味や用法の差異が小さい同音異義語について顕著である．そのため，これら誤りを自動検出し再確認を促すシステムは教育的な視点からも重要性を高めている．

本論文では日本語文中に生じる同音異義語の誤りを検出する手法を提案する．言語モデル BERT を中心とした事前学習モデルにより，大規模コーパスによって事前学習した言語モデルを様々な言語タスクに適用する有効性が示された．本手法では，この BERT の持つ言語空間を用いて同音異義語の誤り検出を行う．BERT の一般的な利用形式は各言語タスクに合わせて，関連するデータにより再学習させる形式である．ここでは Zero-shot 学習の形式をとりいれることで，再学習を経ずに検出する手法を提案する．これにより機械学習の学習コストを抑えつつ，同音異義語の誤り検出について良好な結果が得られることを期待する．

1.2　同音異義語の問題

本論文で用いる同音異義語の定義と問題を説明する．同音異義語は，読みが同じで意味の異なる同音語に分類される．同音語の種類を表 1.1 に示す．

表 1.1: 同音語の種類

	種類	例
同品詞	同音異義語	再建／債権
	表記ゆらぎ	行う／行なう／おこなう
異品詞	動詞／名詞	指す／砂州
	動詞／副詞	上げて／挙げて

ここで同音異義語とは，同音語の単語集合のうち，同品詞かつ表記ゆらぎを含めない単語集合と定義する．本実験では同音異義語の中でも「追求」と「追及」など，同音類義語と呼ばれるような意味の近い語を対象とした．

同音異義語の中には「こうしょう」のように多数の漢字表記を持つものもあるが，ここでは表 1.1 の同音異義語例のように二字ずつの漢字表記を対象として同音異義語の誤り検出を行う．

同音異義語の問題は同音異義語の中から正しい単語を選択する問題とする．誤り検出対象の同音異義語が存在する文を対象とし，その文中において誤った単語が用いられている場合を同音異義語問題の誤りとして検出する．

同音異義語問題の例

「彼との意思疎通は骨が折れる.」

”いし” ＝＞ ”意思” or ”意志”

1.3　本論文の構成

本論文の構成は，はじめに同音異義語問題について関連研究を紹介する（第 2 章）．次に，提案手法で使用した言語モデルの BERT と学習コスト改善のためにとりいれた Zero-shot 学習について順に説明する（第 3 章，第 4 章）．それらをもとにした同音異義

語の誤り検出手法を提案し，実験の内容とその結果について示す（第 5 章，第 6 章）．最後に，それら結果をうけて考察し，本研究の問題点や今後の課題について結論を述べる（第 7 章，第 8 章）．

第 2 章

関連研究

同音異義語の誤りを検出する既存の手法は，複合語に着目し文字連鎖を用いる手法 [1] や決定リストを用いる手法 [2] [3]，確率的 LSA を用いる手法 [4]，Earth Mover's Distance を用いる手法 [5] などが挙げられる．

2.1 複合語に着目した手法

奥らは複合語を構成する単語は互いに何らかの制約を受けていることに着目し，同音異義語の誤り検出に使用する手法を提案した．

意味的な制約を使用する手法の処理手順を図 2.1 に示す．

同音異義語を含む複合語を入力として，形態素解析により不確定状態の同音異義語部とそれ以外の意味属性部に分割する．意味属性部を意味的制約辞書と照合し，その結果を誤り処理に用いる．この意味的な制約を使用する手法は，意味的制約辞書を生成するコストが高いことを問題点として挙げている．その解決として，文書中の文字連鎖を使用する手法 [1] を提案した．この手法の基本的な考え方について奥らは「既存の文書に現れている n 文字連鎖をあらかじめ大量に集めておき，検定対象の不確定単語を含む n 文字連鎖がその中に含まれているかを検証することにより，検定対象の不確定単語が誤りか否かを判定する」と述べている．図 2.2 に文字連鎖を使用する手法の基本的な考え方の模式図を示す．不確定状態の同音異義語部の (n - 1) 文字と，前方または後方に隣接する i 文字を合わせた n 文字連鎖について n 文字連鎖辞書を参照し，その結果を誤り検出に用いる．

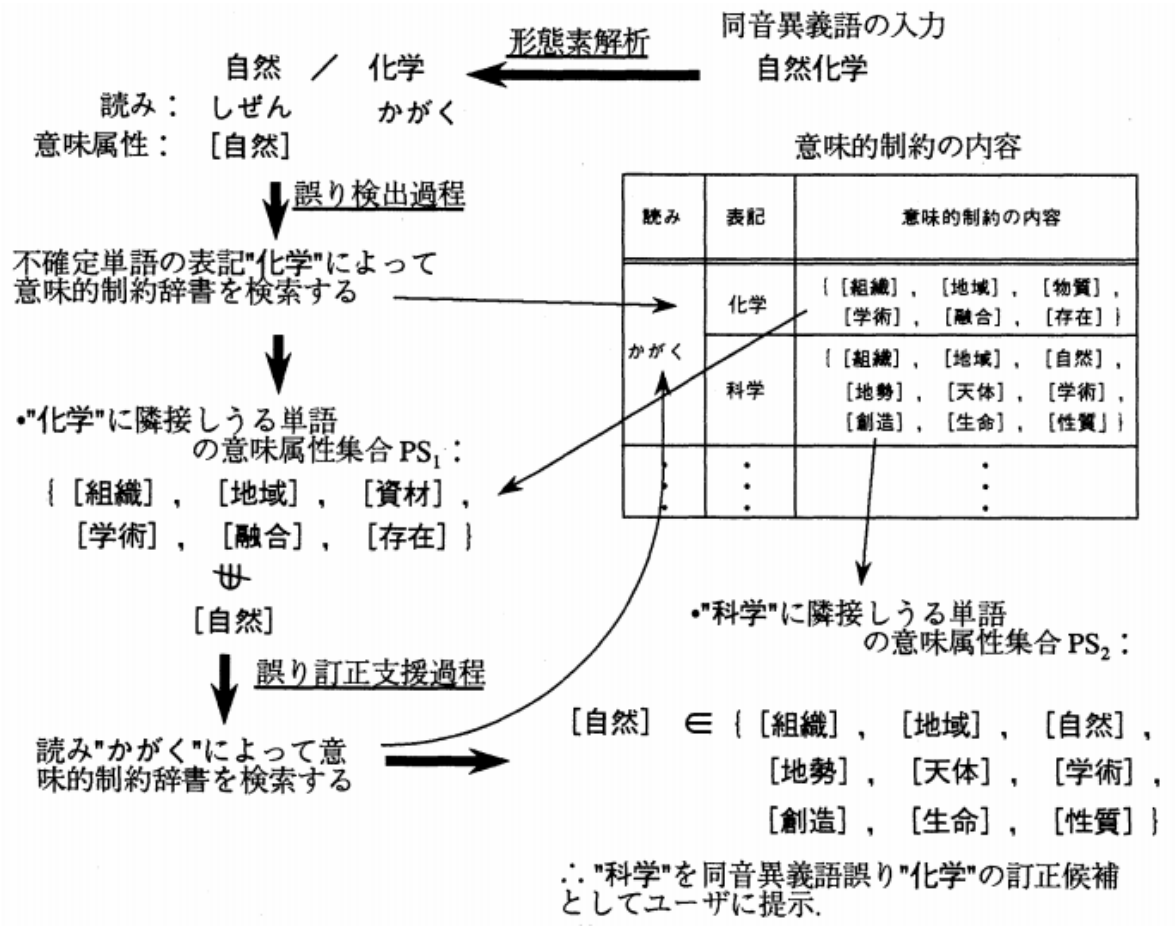

図 2.1: 複合語と意味的制約に着目した手法 (奥ら 1997 [1])

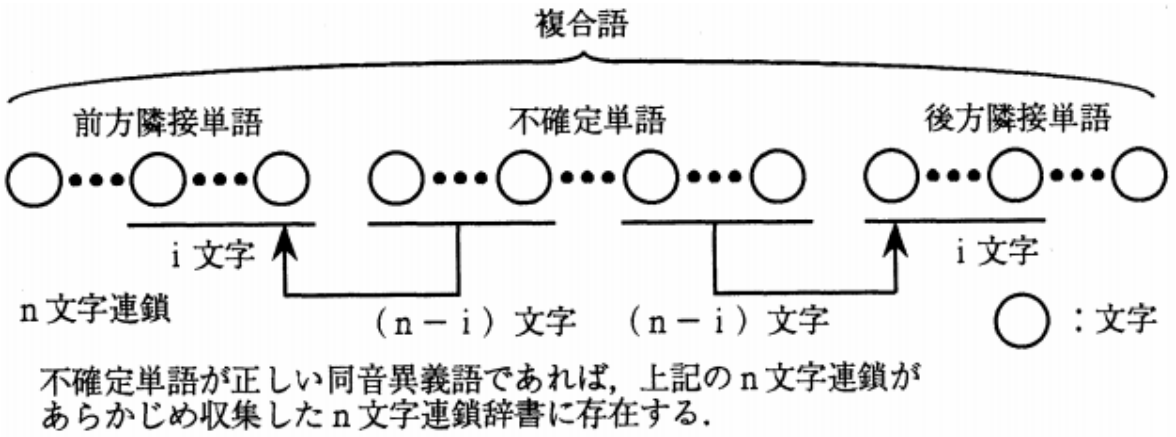

図 2.2: 複合語と文字連鎖に着目した手法 (奥ら 1997 [1])

2.2　決定リストによる手法

新納は「同音異義語問題における平仮名表記を単語，漢字表記を語義ととらえれば，同音異義語問題は語義選択の問題と等価」と解釈することで，従来より語義選択問題に適用されている決定リストを同音異義語問題に適用する手法を提案した．

決定リストはクラス分類手法の一つで確率モデルにより作成される．語義選択をクラス分類とすることで，語義選択問題に使用される．ある単語の語義を選択する際，文脈中で共起する単語を証拠と設定する．それら証拠が語義を決定する予測力を求め，予測力の高い順に並べたリストが決定リストである．「こうしょう」の決定リストの例を表 2.1 に示す．

表 2.1: 決定リストの例

証拠	予測力	クラス
外交	2.22	交渉
理想	1.56	高尚
役場	1.54	公証
学校	1.43	校章
示談	1.22	交渉
…	…	…

予測力は，その証拠 $evidence_i$ のもとで，語義 $sense_a$ が選ばれる確率と語義 $sense_b$ が選ばれる確率との対数尤度比で表される．

$$\log(\frac{P(sense_a|evidence_i)}{P(sense_b|evidence_i)}). \tag{2.1}$$

単に決定リストを同音異義語問題に適用すると，その決定リストは直感的な判断と乖離するため，新納は証拠の重み付けに複合語の情報を使用する手法 [2] と表記情報をデフォルトの証拠とする手法 [3] を提案した．

2.3　確率的 LSA による手法

三品らは記事全体を大域的な情報として用いるために，確率的 LSA を複数の unigram モデルの混合モデルと捉え使用する手法 [4] を提案した．

確率的 LSA(Probabilistic Latent Semantic Analysis 確率的潜在意味解析) において，文脈 h を条件とする単語 w の確率 $p(w|h)$ は次式で与えられる．

$$p(w|h) = \sum_{t=1}^{m} p(t|h)p(w|t). \tag{2.2}$$

t は unigram のモデル番号，m は混合数，$p(t|h)$ は文脈 h における t 番目の unigram の重み，$p(w|t)$ は t 番目の unigram における w の確率である．式 2.2 はある文脈において，同音異義語のどの単語が出やすいかの確率を与える式と解釈できる．

各 unigram モデルを求める際は EM アルゴリズムを用い，以下のような訓練データ D の尤度を最大化し学習する．

$$\mathcal{L}(D;\theta) = \sum_{w} \sum_{d \in D} n(w,d) \log \sum_{t} p(w|t)p(t|d). \tag{2.3}$$

$n(w,d)$ は記事 d 中の単語 w の出現頻度，$p(t|d)$ は記事 d における t 番目のモデルの重みである．この他に三品らは deterministic annealing 法による局所最適解の回避や，変分ベイズ学習による過適応の回避を行っている．

以上から三品らの手法における単語 w の尤度 $L(w)$ は式 2.4 と表現される．

$$L(w) = \frac{p_{PLSA}(w|h_G)}{p_{uni}(w)} p_{ngram}(w|h_L). \tag{2.4}$$

$p_{ngram}(w|h_L)$ は三品らがベースラインとした ngram の手法と同様であり，$p_{PLSA}(w|h_G)$ は PLSA によってモデル化された大域的出現確率を示す．

2.4　EMD による手法

河原らは EMD(Earth Mover's Distance) を用いる手法 [5] を提案した．

EMD は分布間の距離を表す尺度で，類似画像検索の分野などで用いられている．分布間の距離の計算を輸送問題とし，最適な輸送コストを用いて定義する．EMD の算出はま

ず需要地 P と供給地 Q を設定し，それぞれを特徴量と重みのベクトルで表現する．それら特徴量間の輸送コスト $cost_{ij}$ を決め，総輸送コストが最小となる輸送フロー f_{ij} を決定し，分布 P，Q 間の EMD を定める．

$$EMD(P,Q) = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} cost_{ij} f_{ij}}{\sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij}}. \tag{2.5}$$

この EMD の需要地 P を判別対象を含む文，供給地 Q を仮名文字列 h に対する変換候補 $h_1, h_2, \cdots h_n$ とし，それぞれに特徴量と重みを持たせることで同音異義語問題に適用している．

特徴量は大域的情報として文書に出現する名詞，動詞，形容詞の単語を用い，局所的情報として対象同音異義語の直前，直後に出現する単語の品詞情報の組み合わせを用いている．$cost_{ij}$ は品詞情報の合致や単語の出現文書数を用いて算出する．特徴量の重みは tf-idf をもとにしている．

第 3 章

BERT

BERT (Bidirectional Encoder Representations from Transformers) は双方向の Transformer [6] を用い，大規模データによって事前学習された言語モデルである．

この章では，はじめに BERT の中心概念となる Transformer の概要にふれる（第 1 節）．次に BERT の概要と構造を説明し（第 2 節），事前学習のタスクの MLM と NSP について説明する（第 3 節，第 4 節）．その後，本実験で用いた日本語版 BERT について概要を述べる（第 5 節）．

3.1 Transformer

この節では, BERT の “ERT(Encoder Representations from Transformers” にあたる Transformer の概要と構造を説明し，Encoder の概観をみる（第 1 項）．Transformer は二つのサブレイヤーを主に用いている．一つ目のサブレイヤーは Multi-Head Attention である．Multi-Head Attention を理解するために，まず Attention を説明する（第 2 項）．その後，Attention を複数もたせた Multi-Head Attention について述べる（第 3 項）．二つ目のサブレイヤーである position-wise feed-forward networks については一般的な feed-forward networks とほぼ同様のものである．

3.1.1 Transformer の概要と構造

Vaswani らは Transformer [6] という翻訳モデルを提案した．提案の背景はテキスト処理に RNN や CNN を用いる際の問題解決にある．RNN は並列計算に向かず，学習コストを高めてしまう．CNN は長文の依存関係を保持する能力に問題がある．これら解決

のため，Transformer では Recurrent 層や Convolution 層の代わりに Attention 層を用いることが特徴となる．Transformer の全体像は Encoder-Decoder モデルである．そのモデル構造を図 3.1 に示す．

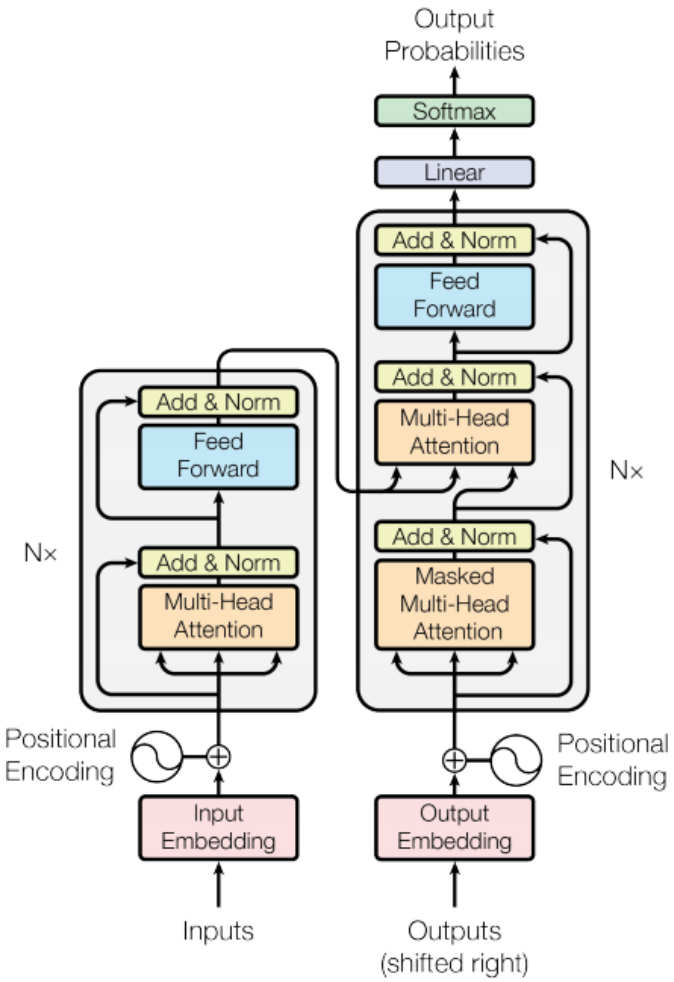

図 3.1: Transformer のモデル構造 (Vaswani ら 2017 [6])

図 3.1 の左側が Encoder，右側が Decoder となっている．どちらにも灰色で囲まれたメインレイヤーがあり，このレイヤーを N 回スタックさせている．Transformer では N=6 としている．このメインレイヤーは，この後の BERT の説明などで Transformer ブロックと称して用いる．

Encoder の領域における処理は，まず入力 (Inputs) を埋め込み表現 (Input Embedding) にし，位置情報 (Positional Encoding) を加える．次のメインレイヤーは二つのサブレイヤーで構成されている．一つ目のサブレイヤーとなる Multi-Head Attention 層に入り，residual connection を加え (Add)，正規化する (Norm)．二つ目のサブレイ

ヤーは一般的な position-wise fully connected feed-forward network である．こちらも residual connection を加え，正規化する．Encoder 領域の大まかな処理は以上である．

Decoder の領域における処理は，まず出力 (Outputs) を埋め込み表現 (Output Embedding) にし，位置情報 (Positional Encoding) を加える．次のメインレイヤーは三つのサブレイヤーで構成されている．一つ目のサブレイヤーとなる Masked Multi-Head Attention 層に入り，residual connection を加え，正規化する．二つ目のサブレイヤーとなる Multi-Head Attention 層は前のサブレイヤーからの出力と Encoder 領域からの出力，二つの出力を入力とする．その後，residual connection を加え，正規化する．三つ目のサブレイヤーは一般的な position-wise fully connected feed-forward network である．こちらも residual connection を加え，正規化する．メインレイヤーを抜けた後に Linear と Softmax により出力確率となる．Decoder 領域の大まかな処理は以上である．

3.1.2　Attention

先述の通り，Transformer は多数の Attention 層を用いている．Attention の求め方を式 3.1，イメージを図 3.2 として示す．

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V. \tag{3.1}$$

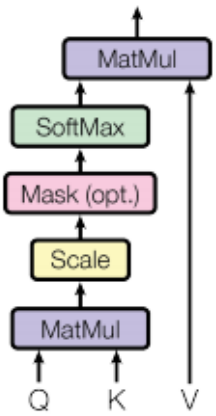

図 3.2: Attention 算出のイメージ (Vaswani ら 2017 [6])

入力は query と key-value のペアを表す Q，K，V のベクトルである．Attention 関数はこの三つをもとにして出力ベクトルへマッピングする処理と言える．

QK^T の行列積を $\sqrt{d_k}$ によりスケーリングし，softmax をとる．その結果と V で行列

積をとったものが出力となる．$\sqrt{d_k}$ の d_k の K の次元数を表す．Q，K の次元数は同じなので，Q の次元数ともとれる．

より具体的な処理として，3 単語を埋め込み表現にした a^1，a^2，a^3 が Attention の算出により b^1，b^2，b^3 となる処理過程を図 3.3 として示す．

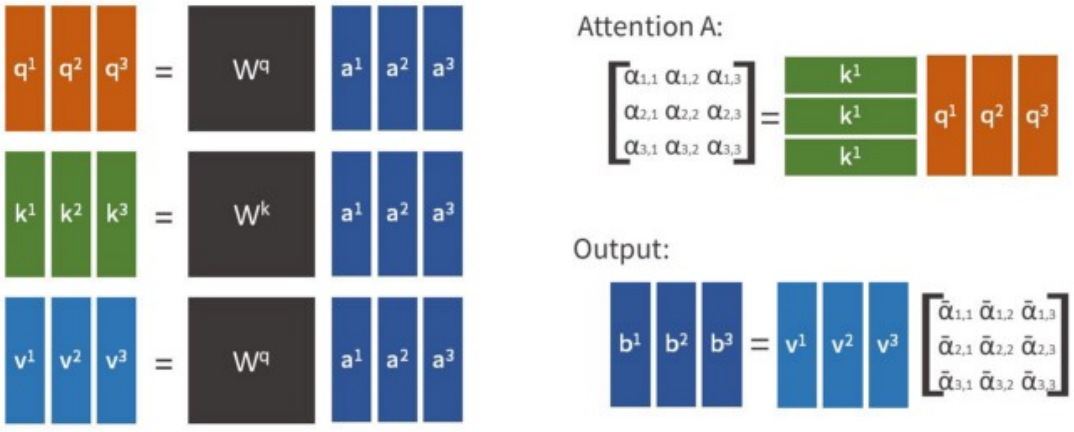

図 3.3: 埋め込み表現 a^1，a^2，a^3 から b^1，b^2，b^3 となる過程[*1]

埋め込み表現 a^1，a^2，a^3 は query，key，value に対応した重み W^q，W^k，W^v により，それぞれ q^1，q^2，q^3，k^1，k^2，k^3，v^1，v^2，v^3 となる．これらが先述の Q，K，V のベクトルである．重み W^q，W^k，W^v が層内で学習される変数行列である．その後，k^1，k^2，k^3 を転置し q^1，q^2，q^3 と行列積をとることで Attention と呼ばれる行列 A が得られる．この A と v^1，v^2，v^3 の行列積が a^1，a^2，a^3 に対応する出力の b^1，b^2，b^3 となる．

3.1.3　Multi-Head Attention

Multi-Head Attention は複数の Attention 行列を Attention 層に持たせることにより，単一の Attention を上回る表現力を獲得するために用いられる．Vaswani らは，単一の 512 次元の Attention を用いるよりも 64 次元の Attention を 8 つ用いて，結果を結合したほうが良い結果になると述べている．

[*1] https://medium.com/lsc-psd/%E8%87%AA%E7%84%B6%E8%A8%80%E8%AA%9E%E5%87%A6%E7%90%86%E3%81%AE%E5%B7%A8%E7%8D%A3-transformer-%E3%81%AEself-attention-layer%E7%B4%B9%E4%BB%8B-a04dc999efc5

Multi-Head Attention の求め方を式 3.2，イメージを図 3.4 として示す．

$$\begin{aligned} MultiHead(Q,K,V) &= Concat(head_1,\ldots,head_h)W^O, \\ where\ head_i &= Attention(QW_i^Q, KW_i^K, VW_i^V). \end{aligned} \tag{3.2}$$

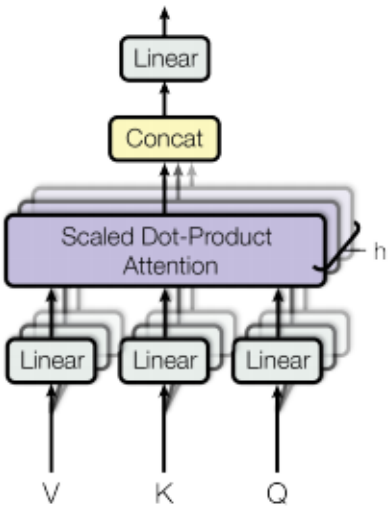

図 3.4: Multi-Head Attention 算出のイメージ (Vaswani ら 2017 [6])

式 3.2 中の h は Multi-Head の head のことで，一つの Attention を意味する．Transformer では $h = 8$ を想定している．モデルの次元を d_{model}，K，Q の次元を d_k，V の次元を d_v として表すと，$d_k = d_v = d_{model}/h = 64$, となる．重みの行列を表す W はそれぞれ $W_i^Q \in \mathbb{R}^{d_{model}\times d_k}$, $W_i^K \in \mathbb{R}^{d_{model}\times d_k}$, $W_i^V \in \mathbb{R}^{d_{model}\times d_v}$, $W_i^O \in \mathbb{R}^{hd_v\times d_{model}}$.

これにより計算上のコストはほぼ同様のまま性能が向上するとされる．

3.2　BERT の概要と構造

Devlin らは BERT [7] という言語モデルを提案した．提案の技術的な背景は，Multi-Head Attention を持つ Transformer ブロックを用いた言語モデルが CNN や RNN に基づくモデルを上回り始めたことが挙げられる．また，それら Transformer ブロックを使ったモデルも単方向の情報を主に用いていたため，双方向性をもたせる改良余地があったことも挙げられる．

ここで，一般的な事前学習モデルを下流タスク（言語処理分野では文書分類など）に適用する方法を確認しておく．

一つ目は feature-based(特徴量ベース）と呼ばれるもので，事前学習モデルによって得

られた表現を，目的のタスクを解くための特徴量として用いる方法である．

二つ目は fine-tuning(微調整) によるアプローチである．当該事前学習モデルを本体として，目的のタスクを解けるよう学習した重みなどを再調整する．

三つ目は教師ありデータからの転移学習が挙げられる．主に画像認識分野で盛んで，大量の教師あり画像データを事前学習し物体の形状や名称をモデルに含ませた上で，目的のタスクで学習する．

BERT は二つ目の fine-tuning によるアプローチをとる．まず大量の教師なしテキストデータによって事前学習モデルを作成する．次に fine-tuning によって様々なタスクに適用する．発表当時，事前学習したモデルの強力さとこの形式により 11 の NLP タスクで SOTA を達成した．

BERT の構造について，Devlin らは Vaswani らが示した Transformer の Encoder を主にそのまま用いたとして参照先に挙げている [6]．本論文では 3.1 節でその内容を先述している．BERT の規模説明のため，BERT が用いる層 (Transformer ブロックなど) の数を”L”，Multi-Head Attention の head の数を”A”として表 3.1 にまとめる．

表 3.1: BERT の規模

model	L	A
Transformer	6	8
BERT BASE	12	12
BERT LARGE	24	16

BERT には BASE モデルと LARGE モデルの二種類があり，そのどちらも Transformer の規模を上回っている．

BERT の入出力について，Devlin らが説明に用いた図を模したものを図 3.5 として示し，これをもとに BERT の入出力について説明する．

BERT の入力表現はラベルの無い二文をトークン化したものと特殊トークンを合わせた一つのシーケンスとなる．特殊トークンは [CLS] と [SEP] に表現され，[CLS] はシーケンス最初に設置されるトークンであり，[SEP] は文の区切りに設置されるトークンである．これらの各トークンに対し，シーケンス中の位置と二文のどちらに属するかの情報を合わせ，埋め込み表現とする．この埋め込み表現を Transformer ブロックを中心とした多層で学習し，その結果を MLM や NSP に用いて下流タスクの精度を高めている．

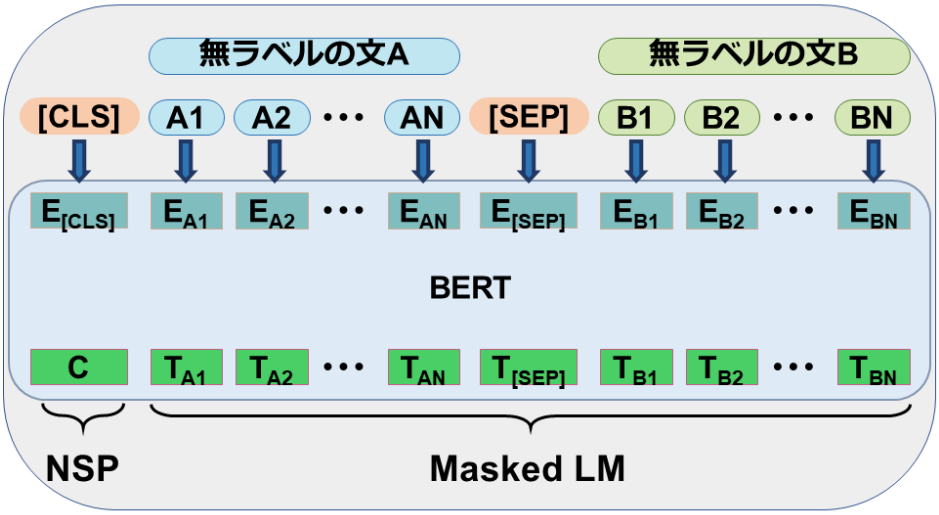

図 3.5: BERT の入出力

3.3　MLM

双方向性獲得の問題点は，多層処理の際に間接的に自分の情報を取得できてしまうことにある．例えば 1 層目で隣に自分の情報を提供した場合，2 層目で隣から渡される情報には自分の情報が含まれている．この問題点に対処し予測力を高めるために，ある割合の入力トークンをランダムにマスクし，それらマスクされたトークンを予測する処理を盛り込む．このマスク処理は空欄補充問題のような処理といえる．文献により cloze タスクとも呼ばれる処理である．

このようなマスク処理を含むモデルを MLM (Masked Language Model) と表現する．

BERT はこの置換割合を 15% としている．置換対象のうち，全てを [MASK] トークンに置き換えると事前学習と fine-tuning の間の齟齬が強く出てしまうため以下のような緩和処理を加えている．

この置換の対象となったトークンのうち，80% を [MASK] トークンに置き換える．10% はランダムなトークンと置き換える．残りの 10% は変更されていない，もとのトークンのままにする．つまり，実質的に [MASK] トークンとされるのは全体の 12% である．

それら [MASK] されたトークンを交差エントロピー誤差をもとに予測するタスクを行う．最終的な [MASK] トークンの埋め込み表現はソフトマックスを通じて各単語と結び

つくこととなる．このタスクにより従来の事前学習手法に比べて，文脈の前後を捉えた双方向性を獲得している．

3.4　NSP

言語モデルが想定する下流タスクには質問応答や自然言語推論など，文と文の関係理解を要するタスクがある．

NSP (Next Sentence Prediction) は，言語モデリングで得られない二文間の関係を BERT に理解させるタスクである．

学習に用いられる単一のコーパスから，シーケンス用の文 A と文 B を選ぶ際，文 B は文 A に連続する文と非連続の文の 2 パターンが半々で選ばれる．非連続の文はコーパス中からランダムに選ばれる．この連続・非連続に二値化された，次の文を予測するタスクが図 3.5 の左下に記される埋め込み表現 C をもとに行われる．これにより二文間の関係の理解力を高め，関連する下流タスクの精度を高めている．

3.5　日本語版 BERT

BERT は英語を対象にモデリングされているため，日本語や中国語のような言語を対象とする場合は入力用に処理を加えたトークン化を要する．主なトークン化の方法は二つある．一つ目は従来の形態素解析器を用いてから，サブワードにするために BPE（Byte Pair Encoding）などのアルゴリズムを用いる方法である．二つ目は SentencePiece[*1] という，ニューラル言語処理向けのトークナイザを用いる方法である．

現在公開されている日本語版 BERT を表 3.2 として示す．モデルサイズのバージョン，BPE の有無バージョン，WWM（Whole WordMasking）の有無バージョン，独自の処理などの詳細や不明な点については省略する．

[*1] https://github.com/google/sentencepiece

表 3.2: 日本語版 BERT の種類

モデル	サブワード	トレーニング
東北大[*1]	MeCab-Neologd + BPE	日本語版 wikipedia
京都大[*2]	JUMAN++ + BPE	日本語版 wikipedia
NICT[*3]	MeCab-Juman + BPE	日本語版 wikipedia
Kikuta[*4]	SentencePiece	日本語版 wikipedia
hottoSNS[*5]	SentencePiece	大規模 SNS コーパス
Stockmark[*6]	MeCab-Neologd	日本語ビジネスニュース記事
Laboro[*7]	SentencePiece	独自収集 Web ページ

[1] https://github.com/cl-tohoku/bert-japanese

[2] http://nlp.ist.i.kyoto-u.ac.jp/index.php?ku_bert_japanese

[3] https://alaginrc.nict.go.jp/nict-bert/index.html

[4] https://github.com/yoheikikuta/bert-japanese

[5] https://github.com/hottolink/hottoSNS-bert

[6] https://qiita.com/mkt3/items/3c1278339ff1bcc0187f

[7] https://laboro.ai/activity/column/engineer/laboro-bert/

本研究で用いた BERT モデルは東北大の BERT のため補足する．

このモデルは日本語版 Wikipedia でトレーニングされている．コーパスの生成には WikiExtractor[*2]を使用して，日本語版 Wikipedia 記事のダンプファイルからプレーンテキストを抽出している．テキストファイルのサイズは 2.6GB で，約 1,700 万文にあたる．プレーンテキストにノイズの除去や適切な文への分割などいくつかの前処理が施される．

トークン化にはまず形態素解析器 MeCab[*3]が用いられる．辞書は mecab-ipadic-NEologd [8] を用いている．英語版のオリジナル BERT と同様にサブワード化には BPE を用いている．この BPE は SentencePiece のものを用いている．これら処理により語彙サイズは 32,000 となる．

モデルの構造も英語版のオリジナル BERT のベースモデルと同様に 12 層，隠れ層のサイズは 768，head の数は 12 となっている．トレーニングも同様にインスタンスあたりのトークンは 512 で，バッチあたり 256 インスタンスとなり，100 万回のトレーニングステップを行っている．

[*2] https://github.com/attardi/wikiextractor
[*3] https://taku910.github.io/mecab/

第 4 章

Zero-shot 学習

Zero-shot 学習は，機械が学習していない未知について人のように推察できるようにするための取り組みである．この章では Zero-shot 学習理解のために，まず Few-shot 学習について説明する（第 1 節）．その後，Zero-shot 学習の概要について述べる（第 2 節）．比較に適した機械学習形式が想定される際はそれらを挙げ，特徴を述べる．

4.1 Few-shot 学習

一般的な機械学習は学習コストが課題となる．学習コストはモデル作成のための計算コストや教師あり学習の教師データを人手で作成するためのコストなどが挙げられる．後者の課題に対しては半教師あり学習，教師なし学習，弱教師あり学習といった対策を生んだ．しかし，これらは基本的にデータの量を前提にしている．一般的な機械学習のアプローチを少量データに用いると脆弱なモデルとなるためである．

Few-shot 学習は，この少量データを前提として機械学習を機能させる取り組みである．通常，N-way-K-shot 分類が想定される．N クラスに各 K 個の例がある分類問題で訓練データの全体数は $K \times N$ 個であり，これが Few-shot 学習の few にあたる数となる．4.2 節で説明する Zero-shot 学習は，例となる情報が zero ということになる．

Few-shot 学習を Wang らの一般化 [9] に沿って概要を説明していく．まず，Few-shot 学習は機械学習に含まれるため，機械学習を定義する．ここでは Wang らと同様に「あるタスク T のためのプログラムが，その性能評価値 P に関して，経験 E によって改善される場合，そのプログラムは経験 E から学習している.」と定義する．few-shot の少量データはタスク固有の限られた数のラベル付きデータを意味し，この経験 E にあたる．

この段階では少量データの教師あり学習と同じ様相となる．少量データのモデル脆弱化に対して，Few-shot 学習は事前知識という後ろ盾を用いる点が異なる．

事前知識について説明する．これはタスク T を処理するのに適したデータ，モデル，アルゴリズムなど多様な対象が挙げられる．これらの事前知識の頑健性をもとにすることで，少量データに対するモデルの安定性を担保する．この事前知識も先述の機械学習定義の経験 E にあたり，その選定制約の基本は few-shot の持つ情報に触れないことである．

ここで転移学習，領域適応，fine-tuning と Few-shot 学習の区別についてふれる．転移学習はデータの豊富なソースドメイン・タスクから少量データのターゲットドメイン・タスクへ知識を転移する点で発想自体似通っている．Few-shot 学習は知識の転移について，転移学習よりも広い概念となる．領域適応は転移学習の一種と考えられ，ソースとターゲットのタスクが同じでドメインが異なるものと考えられる．fine-tuning は事前知識に事前学習モデルを用いたものと考えられる．ただし，事前学習モデルのパラメータを更新する際に少量データによる過剰適合が起きうるため，データ量の視点から注意が必要である．

4.2　Zero-shot 学習の概要

Few-shot 学習の事前知識を前提にすると，機械に未知を推察させられることになる．人は未知を推察する際に形状や色，周囲との関係や伝聞などの経験によって推察する．この経験にあたる情報を機械に知識として提供する．知識には現実を抽象化した形状や言語などの空間を用いることが一般的である．Wang らは Zero-shot 学習について，経験 E にタスク T によらない他のモダリティを含ませる必要があることにも言及している．Zero-shot 学習はタスク T に適した事前知識を選定し，それらをタスク T を解くためにマッピングする処理と言える．

教師あり学習と Few-shot 学習，Zero-shot 学習の違いをクラス分類を例として表 4.1 に示す．

表 4.1: 機械学習の分類

	task T	experience E (教師あり情報)	experience E (事前知識)	performance P
教師あり学習	クラス分類	ラベルありデータ	なし	分類精度
Few-shot 学習	クラス分類	少量ラベルありデータ	事前学習モデル	分類精度
Zero-shot 学習	クラス分類	なし	事前学習モデル	分類精度

第 5 章

提案手法

本章では Zero-shot 学習の形式をとり，BERT によって同音異義語の誤り検出を行う手法を述べる．

Wang らの定義を参考に本手法を表現すると，タスク T はクラス分類としての同音異義語の誤り検出，経験 E は事前知識として事前学習モデル BERT，性能評価値 P は分類精度としての F 値となる．

まず，誤り検出の対象文として同音異義語が含まれる文を入力する．入力文中から対象となる同音異義語部を BERT で [MASK] トークンの役割を持つトークンに置き換える．その後，入力文を全体をトークン化し，事前学習された BERT により入力文全体を埋め込み表現にする．この埋め込み表現となったトークン列をもとに [MASK] トークンの埋め込み表現を BERT に予測させる．予測結果を BERT の尤度順に 10,000 語取得し，対象となる同音異義語の出現順を調べる．出現順の早い，BERT の尤度が高い方を入力文中で用いられるべき同音異義語の予測結果とする．予測結果と入力文の同音異義語が異なる場合に誤りとして検出する．

BERT の本来の利用法は事前学習モデルをタスクに fine-tuning する手順が想定されている．本手法は，この fine-tuning のためにデータを用意するコストや fine-tuning 自体の計算コストを Zero-shot 学習の形式をとることで節約できる．事前学習モデルの学習コストをこの Zero-shot 学習の形式に含めるか否かについては大きなテーマなため，ここでは言及しない．

第 6 章

実験

本章では，本手法の効果をみるために行った実験の内容についてふれる．まず，実験に関連するデータや用いたモデルなどの設定について述べる（第 1 節）．その後，実験の結果を示す（第 2 節）．

6.1 実験設定

実験に用いたデータは 1993 年から 1999 年の毎日新聞の記事内で用いられた 6,393,423[*1]の日本語文データである．表 6.1 にデータに対して行った前処理と文数を示す．

表 6.1: 実験用記事内の総文数と前処理後の文数

内容	文数
オリジナルデータ	6,393,423
400 文字を超える文を削除	6,392,701
平仮名を含まない文を削除	6,333,650

オリジナルデータから作文用紙 1 枚分にあたる 400 文字を超える文は除いた．記事の中には日本語文と言えないものも含まれているため，平仮名を含まない文も除いている．

誤り検出の対象とする同音異義語は表 6.2 に示す 20 組を対象とする．選定の基準は『類義語の研究』で行われた同音類義語の調査 [10] にもとづいている．これは 1962 年に

[*1] 一部重複を除いている．

都立高校三年生 143 人を対象に同音類義語 25 組をテストした結果である．人が識別した結果の一例として大いに参考にした．この中から，毎日新聞記事に登場する文数との兼ね合いや研究当初から対象としていた 2 種類を含めた計 20 組を対象としている．

前処理をした記事の全体データから，対象の同音異義語を含む文を実験用に取り出す．取り出す際は MeCab[*2]を用いて単語の確認をしている．MeCab 用の辞書は mecab-ipadic-NEologd [8] を用いている．取り出した文は同音異義語ごとに偏りがあるため，比較する同音異義語の文数が少ない方に合わせて調整している．この数量調整のための抽出はランダムに文を選ぶようにしている．記事内から取り出した対象単語を含む文数と実験のために抽出した文数についても表 6.2 にあわせて示している．

同音異義語の誤りデータは，記事の内部に記載されている同音異義語について当該箇所だけをもう一方の同音異義語に置き換えることで作成した．誤りデータは各同音異義語の抽出数からの 50% ずつ作成し，それらを合わせたものを実験に用いた．

BERT の事前学習モデルには東北大学から公開されている日本語版 BERT[*3]を用いている．この BERT の設定は「cl-tohoku/bert-base-japanese-whole-word-masking」のモデルを公開されているデフォルトのままで用いている．このデフォルトについては 3.5 節で述べている．

実験結果の評価指標には F 値を用いる．本実験のテストデータは二値分類として偏りが無いよう調整しているが，本研究は対象や条件の追加が想定されるため，正解率や再現率を考慮できるよう F 値を用いている．

事前知識として用いた BERT の事前学習モデルが本実験で対象とした同音異義語について fine-tuning 無しにどの程度の識別するのかを検証するため，テストデータに誤りを含めない，新聞記事のままで予測させる実験を一部の同音異義語を対象に行っている．

[*2] https://taku910.github.io/mecab/

[*3] https://github.com/cl-tohoku/bert-japanese

表 6.2: 実験用データ内の対象同音異義語

単語	記事内	抽出数	実験数	単語	記事内	抽出数	実験数
修了	964	900	1,800	保障	3,414	2,500	5,000
終了	12,665	900		保証	6,674	2,500	
追求	2,825	2,500	5,000	共同	15,406	300	600
追及	9,029	2,500		協同	307	300	
同士	8,143	600	1,200	製作	4,251	2,500	5,000
同志	687	600		制作	6,685	2,500	
競走	838	800	1,600	実体	814	800	1,600
競争	8,962	800		実態	11,875	800	
平行	505	500	1,000	体制	17,453	5,000	10,000
並行	1,741	500		態勢	5,146	5,000	
解放	5,177	2,500	5,000	鑑賞	1,283	600	1,200
開放	4,679	2,500		観賞	666	600	
移動	5,864	1,000	5,000	作成	9,428	800	1,600
異動	1,112	1,000		作製	828	800	
意志	964	900	1,800	異状	399	300	600
意思	9,268	900		異常	6,213	300	
最後	26,739	700	1,400	振動	1,047	100	200
最期	723	700		震動	137	100	
解答	562	500	1,000	不断	192	100	200
回答	10,871	500		普段	3,907	100	

6.2　実験結果

実験の結果と高校生の正解率を表 6.3 に示す.

表 6.3: 実験結果

	実験数	F 値	正答率	高校生
{ 修了，終了 }	1,800	0.952	95.2%	79.5%
{ 追求，追及 }	5,000	0.943	94.4%	63.6%[*1]
{ 同士，同志 }	1,200	0.939	94.0%	72.4%
{ 競走，競争 }	1,600	0.925	92.6%	88.6%
{ 平行，並行 }	1,000	0.906	90.6%	72.4%
{ 解放，開放 }	5,000	0.897	89.8%	76.7%
{ 移動，異動 }	2,000	0.895	89.5%	81.6%
{ 意志，意思 }	1,800	0.792	79.8%	—%
{ 最後，最期 }	1,400	0.774	77.6%	74.4%
{ 解答，回答 }	1,000	0.766	76.5%	—%
{ 保障，保証 }	5,000	0.765	76.9%	82.0%
{ 共同，協同 }	600	0.764	76.7%	62.2%
{ 製作，制作 }	5,000	0.716	71.5%	65.5%
{ 実体，実態 }	1,600	0.699	69.8%	83.4%
{ 体制，態勢 }	10,000	0.635	63.7%	68.2%[*2]
{ 鑑賞，観賞 }	1,200	0.577	57.9%	67.1%
{ 作成，作製 }	1,600	0.516	51.8%	68.7%
{ 異状，異常 }	600	0.511	49.7%	49.3%
{ 振動，震動 }	200	0.508	51.5%	69.8%
{ 不断，普段 }	200	0.490	50.0%	87.9%

[1] 追究との三択

[2] 大勢との三択

実験結果について言及する前に，いくつかの注意点を述べる．

高校生の正解率には一部三択の結果が含まれている．{ 追求，追及 } の項目で，「追究」を含んだ三択，{ 体制，態勢 } の項目で，「大勢」を含んだ三択の結果が用いられている．この三択目の選択肢について松尾らは「使い分けは，明確に意識されていると思われる．」と評しているため，大勢に影響は無いものと思われる．

また，実験の拡張を想定して F 値を用いているが，現状の 50% の誤りデータを用いるという条件上は正答率とほぼ同様なので，高校生の正答率と比較する上でもここでは正答率を中心に述べる．また，問題自体が異なるため高校生の正答率は参考程度であり，一般的な感覚の「人」と，参考にした「高校生」の結果は分けて述べる．

実験の結果を順に見る．まず，全体的に人よりも得意不得意がはっきりと別れていることが見て取れる．得意な同音異義語の誤り検出精度については難度を考慮しても人より優れている可能性がある．一方，{ 不断，普段 } のように人が識別しやすい語について，大きく精度を崩しているものも見受けられる．

個別の結果にうつる．最も正答率の高い結果は { 修了，終了 } の 95.2% である．問題によるが，一般的な人の正答率を上回る可能性がある．検出に失敗した例を表 6.4 に示す．

表 6.4: 修了，終了の予測誤りの例

予測結果	番号	文
正文を誤文と予測	文 1	現在のように，義務教育修了という条件のない時代.
	文 2	週１～２回のペースで開き，約２カ月で修了する.
誤文を正文と予測	文 3	さらに大学院終了間際，今度は腎臓（じんぞう）結核で片方を摘出するなど闘病の連続.
	文 4	半年が１期で，終了すると２単位で，３年間行う計画.

表 6.4-文 1 は教育を修めたので「修了」を用いるのが正しい．表 6.4-文 2 は「なに」を「しゅうりょう」したのか不明なため，どちらも正解として良い例である．表 6.4-文 3 は大学院を「修了」と考えるべきで誤っている．表 6.4-文 4 は「単位」から学問と推察し，「修了」を用いるべきだが難題である．

{ 異状，異常 } の結果についてもふれる．この結果は提案手法と高校生のどちらも良い結果と言えない．人にとっても機械にとっても難題で，ほぼ意味は同じと言って良い．

用法として,「異状」は名詞で用いるが,「異常」は名詞のほかに「異常な」などの形容動詞として用いられることもある．どちらも名詞が想定される場合，状態性の認識による．

問題点として先述した { 不断，普段 } の結果は 50.0% で二値分類の意味がない．高校生の正答率は 87.9% であり，使い分けが明らかに意識されている．検出に失敗した例を表 6.5 に示す．

表 6.5: 不断，普段の予測誤りの例

予測結果	番号	文
正文を誤文と予測	文 1	言葉を自在に操るための不断の努力が欠かせないのである．
	文 2	問題を招いている制度や規則の不断の見直しが重要だ．
誤文を正文と予測	文 3	憲法は普段の努力によって実現しなければならない．
	文 4	だからと言って普段の努力を怠るのでは目標はさらに遠のく．

この例を見て分かるように,「不断」と「努力」の関係を捉えきれていない．「不断」の利用場面で非常に多いフレーズである．「努力」が文中に無い場面では,「不断」と「普段」は頻度状態表現としてほぼ似通っており，話者の意識などを理解しなければどちらでも当てはまる場面が多い．表 6.5-文 2 と文 4 は話者が「普段」を意識していれば「不断」で無くても良い．

実験に用いたデータ数による影響も考えられる．実験データの少ない同音異義語の組について，少なくとも一方が新聞記事でさえ用いられにくい単語ということである．これにより機械が学習しきっていない．一方で人はあまり見ない量のデータでも，十分学習しているという知見が得られる．

機械が学習しきらない点を検証するため，精度の悪かった同音異義語の組を中心に追加の実験を行っている．誤りを含めない新聞記事のままで対象単語を予測させた一部の結果を表 6.6 に示す．BERT の 10,000 語以内に予測されなかった場合は「誤り」として一律に処理している．

{ 不断，普段 } は「不断」を予測していない．{ 不断，普段 } のように，実験結果がほぼ 50% となった { 作成，作製 }，{ 異状，異常 }，{ 振動，震動 } については一方のみを予測している状態におちいっている．

正答率 77.6% の { 最後，最期 } は記事内の登場数に約 26,000 の差があるが一方のみを予測している状態にはおちいっていない．しかし,「最後」の識別力が結果を高めてい

表 6.6: 誤りを含めない識別結果

	記事内数	正	誤	正答率
不断	192	0	192	0%
普段	3,907	3,887	20	99.5%
鑑賞	9,268	8,227	1,041	88.8%
観賞	666	131	535	19.7%
最後	26,739	26,378	361	98.6%
最期	723	404	319	55.9%
修了	964	891	73	92.4%
終了	12,665	12,404	261	97.9%

るだけで,「最期」の方は 55.9% と識別していない.

{ 修了，終了 } は使い分けが十分されている.

検出精度の悪い同音異義語の組については，一方が登場しないような埋め込み表現になっている可能性がある.

第 7 章

考察

まず，選択した同音異義語の種類については難度が高く，解釈によってはどちらも正解という場面が多かった．先述した { 異状，異常 } は最たる例である．あえて選ぶならば狭義な方という場面も多く，包含関係を事前知識に含めなければ解決し得ない．

次に，人よりも同音異義語の識別に得手不得手が現れた点については { 不断，普段 } の実験結果に見るように，不得手については一方の語を予測しないためである．人ならば分かるものに手を付け，分からないものは運に任せるという行動をとるため 50% 付近になることはあまりない．また，{ 修了，終了 } の結果に見るように，機械は厳密にルールに沿って答えている．人は経験による解釈が生じるため，どちらも正解という迷いが生まれると推察される．

一方の語のみ予測する現象は過剰適合など機械学習でよく見られる．アルゴリズムやデータによって改良できる可能性がある．本実験は日本語版 Wikipedia コーパスを用いた事前学習モデルを用いている．このデータについて質と量の検証が必要である．Wikipedia は，一般善意の集合体なため新聞記事のように校閲などの処理は基本的にない．質として疑問が残る点となる．また，大規模データを前提とする事前学習モデルに，本実験のような単語ごとの量的な調整は行われない．このような事前学習データの質と量の問題が，機械学習の過剰適合に繋がったのではないかと推察する．

Zero-shot 学習の形式については繰り返しになるが，事前学習モデルの学習コストを Zero-shot 学習の形式に含めるか否かについて大きなテーマなため，ここでは言及しない．その上で本手法を行った所感として，学習コストが課題とならなかった点だけ述べておく．

また，この形式を用いたことで fine-tuning によって見失われがちな汎用状態の事前学

習モデルの問題点が明らかになる有用性があった．

考察の最後に，本実験では機械がある語を予測しない事象や，特徴的な狭義表現がありながら広義の表現を多用する事象がみられた．これらは言語の豊かさを損なう事象である．人が用いて新しい語を作り，人が用いず語が廃れるのは自然であるが，機械が助長してはならないと考える．このような点からも，語を機械が識別する取り組みの重要性を考えさせられた．

第 8 章

結論

本研究では Zero-shot 学習の形式により機械学習の学習コストを抑えつつ，事前学習モデル BERT を同音異義語の誤り検出に利用することを目的とした．同音異義語は人でも識別に難ある同音類義語と呼ばれるような語を対象とした．

結果は得手不得手あるが，平均して高校三年生ほどの識別能力が得られた．また，本手法中で学習コストが負担になることは無かった．実験結果の一部に，識別対象の一方のみを予測する状態が見られた．これは fine-tuning を経ないために汎用状態の事前学習モデルの問題点が現れたものと推察される．

今後の課題は二点挙げられる．

一点目は，汎用状態の事前学習モデルの改良である．BERT の事前学習のために用いられた日本語版 Wikipedia コーパスの傾向を調査し，実験結果との相関性を検証する必要がある．この検証結果をもとに事前学習モデルに対するデータの量と質を調整し，事前学習モデルをどの程度まで再構築する必要があるのか検討を進めたい．

二点目は，Zero-shot 学習の形式の改良である．本研究は Zero-shot 学習の事前知識を BERT のみで行った．事前知識を複数採用し，集合知の状態とした場合の実験を試みたい．これにより本実験の極端な結果について緩和や改善が期待できる．また，事前知識に設定するモデルやデータ，アルゴリズムの組み合わせによる効果もあわせて確認したい．

謝辞

本研究を進めるにあたり，多くのご指導を頂きました指導教員の新納教授に深い感謝の意を表します．また，多くのご意見，ご指摘を頂いた自然言語処理研究室の皆様に心より感謝申し上げます．

参考文献

[1] 奥雅博, 松岡浩司. 文字連鎖を用いた複合語同音異義語誤りの検出手法とその評価. 自然言語処理, Vol. 4, No. 3, pp. 83–99, 1997.

[2] 新納浩幸ほか. 複合語からの証拠に重みをつけた決定リストによる同音異義語判別. 情報処理学会論文誌, Vol. 39, No. 12, pp. 3200–3206, 1998.

[3] 新納浩幸ほか. 表記情報をデフォルトの証拠として用いた決定リストによる同音異義語の誤り検出. 情報処理学会論文誌, Vol. 41, No. 4, pp. 1046–1053, 2000.

[4] 三品拓也, 貞光九月, 山本幹雄ほか. 確率的 lsa を用いた日本語同音異義語誤りの検出・訂正. 情報処理学会論文誌, Vol. 45, No. 9, pp. 2168–2176, 2004.

[5] 河原直人, 梅澤猛, 大澤範高ほか. E-016 earth mover's distance を用いた同音異義語判別 (e 分野: 自然言語・音声・音楽). 情報科学技術フォーラム講演論文集, Vol. 12, No. 2, pp. 217–218, 2013.

[6] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[8] 奥村学佐藤敏紀. 単語分かち書き辞書 mecab-ipadic-neologd の実装と情報検索における効果的な使用方法の検討. 言語処理学会第 23 回年次大会 (NLP2017), pp. NLP2017–B6–1. 言語処理学会, 2017.

[9] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. Generalizing from a few examples: A survey on few-shot learning. *ACM Computing Surveys (CSUR)*, Vol. 53, No. 3, pp. 1–34, 2020.

[10] 松尾拾, 西尾寅弥, 田中章夫. 類義語の研究. 国立国語研究所, 1965.