令和 5 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 を利用した中国語漢字の読み曖昧性解消

所属　新納研究室

著者　TANG BIN (22NM737N)

指導教員　新納浩幸教授

令和 6 年 1 月 28 日 (木)

令和 5 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 を利用した中国語漢字の読み曖昧性解消

著者

TANG BIN (22NM737N)

指導教員

新納浩幸教授

論文要旨

近年、音声アシストアプリケーションなどで使用される音声合成技術への需要が高まっている。ただし、中国語においては、日常的に使用される漢字の約 10.7% [1] に読みの曖昧性が存在しており、これが技術的な課題として挙げられる。読みの曖昧性とは、同一の漢字が異なる意味によって異なる発音を持つ現象を指す。本研究では、BERT モデルを基礎とし、その自己注意メカニズム及び全結合層の改良を通じて、中国語テキスト内の多音字認識問題に対処する。BERT モデルの重要な要素の最適化を行い、多音字認識タスクへの適用を可能にした。また、本稿では、BERT の訓練プロセスに変更を加え、定向マスクと追加学習を組み合わせる手法を提案し、識別精度の向上を目指す。中国語ニュースの前処理段階では、多音字の正しい発音を直接多音字の後に挿入する。訓練プロセスにおいては、多音字に対する定向マスク操作を行い、モデルの訓練効率を向上させる。テスト段階では、モデルが異なる多音字をどの程度正確に識別できるかを検証し、提案手法の有効性を論証する。最終目的は、多音字の発音を正確に識別することである。実験結果としては、全体の正答率が 92.37% に達し、さらに、モデルの精度向上のために実験パラメータの調整や対照実験を行い、品詞タグを付与して追加学習を実施し、パラメータがモデル性能に及ぼす影響を検討した。従来の手法を簡素化し、読みの曖昧性解消の目的を達成することができた。

Master’ s Thesis in Scholastic 2023, Major in Computer and Information Sciences,Granduate School of Science and Engineering, Ibaraki University

Eliminating the Ambiguity of Chinese Character Pronunciation Using BERT

Author : TANG BIN (22NM737N)

Adviser : Prof. Hiroyuki Shinnou

Abstract

In recent years, there has been a growing demand for speech synthesis technology, particularly in applications such as voice assistant applications. However, in Chinese, approximately 10.7% [1] of commonly used characters have reading ambiguities, which pose technical challenges. Reading ambiguity refers to the phenomenon where the same character has different pronunciations due to different meanings. In this study, we address the issue of polyphone recognition in Chinese text using a BERT model as a foundation, by optimizing its self-attention mechanism and fully connected layers. We optimized crucial elements of the BERT model and applied them to the task of polyphone recognition. Additionally, we propose a method combining directional masking and additional training in the BERT training process to improve classification accuracy. In the preprocessing stage of Chinese news, we directly insert the correct pronunciation of polyphones after the polyphones themselves. During the training process, we perform directional mask operations on polyphones to improve the efficiency of model training. In the testing stage, we verify how accurately the model can identify different polyphones and demonstrate the effectiveness of the proposed method. The ultimate goal is to accurately identify the pronunciation of polyphones. As experimental results show, the overall accuracy reached 92.37%. Furthermore, we adjusted experimental parameters and conducted control experiments to enhance model accuracy, including assigning part-of-speech tags and implementing additional training, while investigating the impact of parameters on model performance. By simplifying conventional methods, we achieved the goal of resolving reading ambiguities.

目次

第 1 章　序論 8

1.1　中国語読み曖昧性の問題 8

1.2　本論文の構成 9

第 2 章　関連研究 10

2.1　機械学習手法と単語埋め込み表現 10

2.2　RNN 11

2.3　LSTM 11

2.4　transformer 12

2.5　多音字認識の関連研究 14

2.6　伝統的深層学習モデルに基づく手法 14

2.7　事前トレーニング言語モデル（PLM）に基づく手法 15

第 3 章　BERT モデル 19

3.1　前処理 19

3.2　埋め込み層 21

3.3　Transformer 層 22

3.4　事前学習タスク 24

第 4 章　提案手法 26

4.1　デザインのアプローチ 26

4.2　データ処理 27

4.3　モデル構造 30

4.4　モデルの入力と推論 30

第 5 章　実験 34

5.1　データセット 34

5.2　実験設定 37

5.3　実験結果 39

第 6 章　考察 40

第 7 章　結論 42

7.1　まとめ 42

7.2　今後の予定 43

参考文献 46

図目次

2.1 RNN モデル結構 11
2.2 LSTM 結构 12
2.3 Transformer 結構 13
2.4 PDF 結構 16
2.5 g2pW 多音字識別モデル結構 17

3.1 Bert モデル 20
3.2 テキストを数字のマーク列に変換する概略図 21
3.3 単語埋め込みの結構図 21
3.4 単語埋め込みの結構図 22
3.5 マルチヘッド注意機構の構造図 24

4.1 データの前処理プロセス 28
4.2 データ前処理の例 28
4.3 品詞を用いて手法プロセス 29
4.4 前処理プロセス 29
4.5 多音字モデルの構築（左） と attention layers の構築（右） 31
4.6 入力のケース 32
4.7 予測プロセス 33

5.1 各データセットの多音字の統計 36
5.2 各データセットの多音字組の統計 36
5.3 発音のみの手法の損失関数 38
5.4 発音と品詞用いる手法の損失関数 39

表目次

1.1 多音字データの例 9

5.1 多音字データの例 35
5.2 多音字の発音統計 35
5.3 各データセットの統計 36
5.4 実験詳細 37

6.1 実験データ 41

第 1 章

序論

近年、デジタル化及び情報化の進展に伴い、言語処理技術の需要が急激に増加している。しかし、多音字の存在が中国語における文章理解において大きな課題をもたらしている。中国語においては多音字現象が一般的で、一つの文字が複数の発音を持つ現象を指す。この多音字の存在は、歴史的な音声変化、方言の違い、外来語の影響など、多様な要因によるものである。関連研究によれば、多音字は中国語の単語データベースにおいて約 11.7% を占め、これらの多音字は自然言語処理システムに特定の課題をもたらしている。

多音字の発音は、通常、所属する単語、品詞、文脈によって決定され、簡単には識別できない。このため、BERT モデルは双方向の事前学習手法を採用し、文脈情報に基づいて単語の文脈を理解する能力を持つ。これにより、多音字の発音認識タスクを遂行することが可能となった。ただし、自然言語処理のタスクを機械学習のアプローチで解決しようとする場合、訓練データのサーバー負荷が高まるという課題がある。そこで、本研究では、BERT モデルを簡略化し、機械学習の学習コストを抑えつつ、より精確かつ迅速に多音字の発音を認識できる手法を提案し、音声合成及び音声認識技術に新たな解決策を提供する。

1.1　中国語読み曖昧性の問題

多音字とは、中国語において、一つの漢字に複数読み方があること、これらの発音は文字の意味、用法、および品詞と密接に関連しています。また、表 1.1 のように、大部分の多音字は 2 種類だけでなく、3-4 種類の読み方を持つものもよくあり、同じ文字でも異な

る地域や方言では異なる発音が存在する可能性があります。

表 1.1: 多音字データの例

多音字	各多音字の発音					発音数
声	shēng	chéng				2
强	qiāng	jiàng	qiǎng			3
着	zhuó	zháo	zhě	zhāo		4
和	hé	hě	huó	huò	hū	5

1.2　本論文の構成

本論文の構成は以下の通りである。はじめに多音字の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 BERT 言語モデルと提案方法について順に説明する（第 3 章、第 4 章）．それらをもとに実験の内容とその結果について示す（第 5 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 6 章、第 7 章）

第 2 章

関連研究

2.1 機械学習手法と単語埋め込み表現

近年、自然言語処理の分野では、深層学習への移行が始まり、畳み込みニューラルネットワークから始まる深層学習のブームが引き起こされ、多様な新しいニューラルネットワークモデルが自然言語処理のタスクに広く適用されている。自然言語処理の問題を深層学習モデルで解決する際の第一歩は、言語をニューラルネットワークが処理可能なデータ型に変換することである。言語をニューラルネットワークが理解できる数値形式に変換するため、単語埋め込み技術が開発された。2013 年に Tomas Mikolov らによって提案された Word2vec モデルは、単語埋め込みの代表例として NLP タスクに広く適用されている。しかし、これらの単語埋め込みは入力の単語情報を捉えることはできるが、文脈に依存しないため、単語間の意味情報や文法構造、一語多義の状況を捉えることはできない。2017 年には、Transformer アーキテクチャに基づく自然言語処理モデルが登場し、従来のリカレントニューラルネットワークおよびその変種構造を排除し、注意機構を採用した。これにより、自然言語処理の分野は事前トレーニング言語モデルの新時代を迎えた。その後、研究者たちは注意機構を文脈に基づく単語埋め込み学習に適用し、事前トレーニングされた Transformer 言語モデルが注目すべき成果を上げている。例としては、ELMo、ULMFit、GPT、BERT、ENRIE、XL-Net、RoBERTa などがある。

2.2　RNN

再帰型ニューラルネットワーク（Recurrent Neural Networks、RNN）[2] は、時系列データを順次に入力し、データをシーケンスの進行方向に沿って再帰的に処理するニューラルネットワークの一種である。このモデルの構造は図 2.1 に示されている。ニューラルネットワークモデル構造において、入力層、隠れ層、出力層が完全に接続されているが、ノード間の通信がないため、次の単語を予測することが困難であるという問題が存在する。しかし、RNN はこの問題を解決することができる。RNN の構造では、各層間のノードが接続されており、前の情報を記憶し、現在必要な出力計算に利用することが可能である。そのため、自然言語処理において、言語モデリング、品詞タギング、さまざまな時間系列予測などの分野で RNN は一定の成果を上げている。

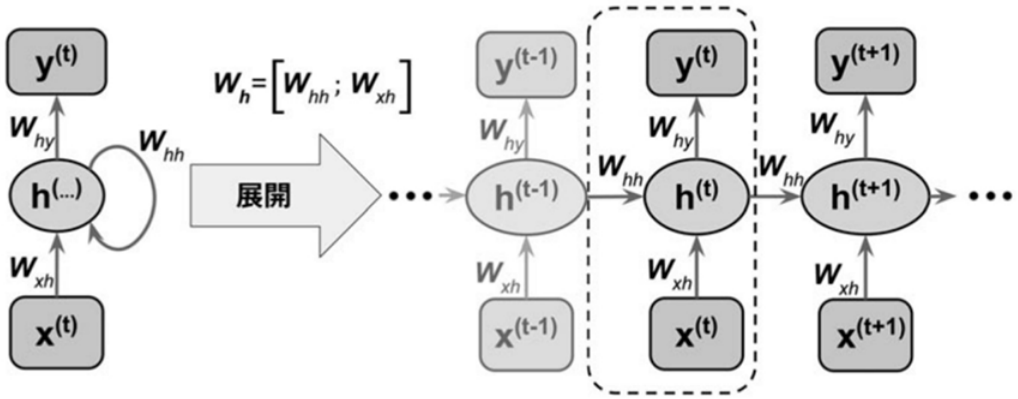

図 2.1: RNN モデル結構

2.3　LSTM

RNN モデルにおける勾配消失及び勾配爆発の問題を解決するために、長短期記憶（Long Short-Term Memory、LSTM）[3] が提案された。LSTM は長期的な依存関係を処理し、文の順序を学習および予測する能力を持つ。これにより、バックプロパゲーションアルゴリズムを用いた計算が時間を要するという問題が解決される。LSTM は、その特有の構造により、情報を長期間にわたって保持し、必要に応じて情報を忘れる機能を有しているため、自然言語処理を含む多くの分野で有効に活用されている。

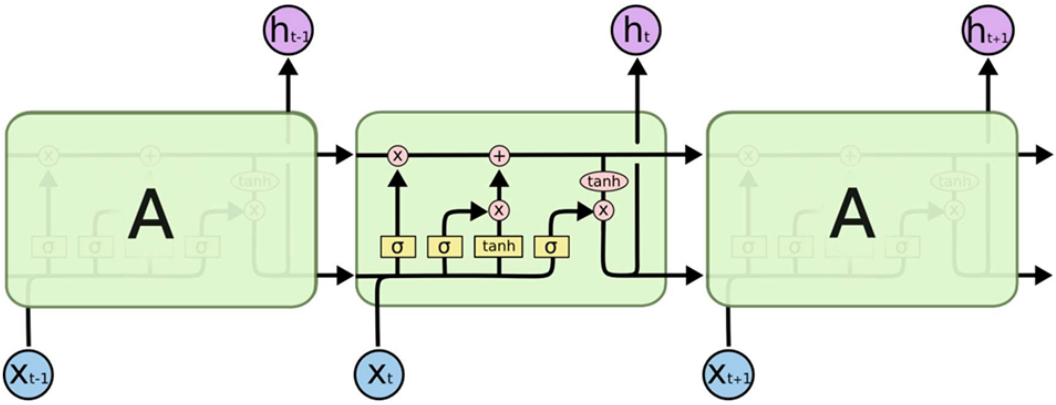

図 2.2: LSTM 結構

図 2.2 に示されている通り、従来の RNN モデルに比べて、LSTM には 2 つの伝達状態が存在する。一つはセル状態（cell state）であり、もう一つは隠れ状態（hidden state）である。LSTM モデルでは、情報をセル状態に移動させたり追加したりするために、「ゲート」構造を採用している。この「ゲート」は情報に対して選択的であり、sigmoid ニューラルネットワーク層とビットワイズ乗算演算から構成されている。LSTM モデルは、遺忘ゲート、入力ゲート、出力ゲートの 3 つを使用して、セル状態の保持と制御を行う。遺忘ゲートは有用な情報を選択的に保持し、不要な情報を破棄する。入力ゲートは更新すべき値を決定し、sigmoid 層と tanh 層を使用して状態を更新する。出力ゲートは出力すべき情報を決定する。これらのゲート機能により、LSTM は従来の RNN モデルに比べて情報の長期保持と管理が可能となっている。

2.4　transformer

Google チームにより提案された Transformer モデル [4] は、神経ネットワークモデルの一つで、最初は機械翻訳の分野に適用され、後に自然言語処理の多様なサブタスクに広く応用されて優れた成績を収め、この領域において不可欠な古典的なモデルとして認識されている。BERT モデルは、Transformer モデルを基礎として派生した事前トレーニング言語モデルである。Transformer モデルの詳細な構造は、図 2.3 に示されている

Transformer モデルは、完全な注意機構に基づくエンドツーエンドのアーキテクチャを採用している。このモデルは、主にエンコーダーレイヤとデコーダーレイヤの 2 つの部分から構成される。エンコーダーレイヤは、図 2.3 に示された左側の点線のボックス

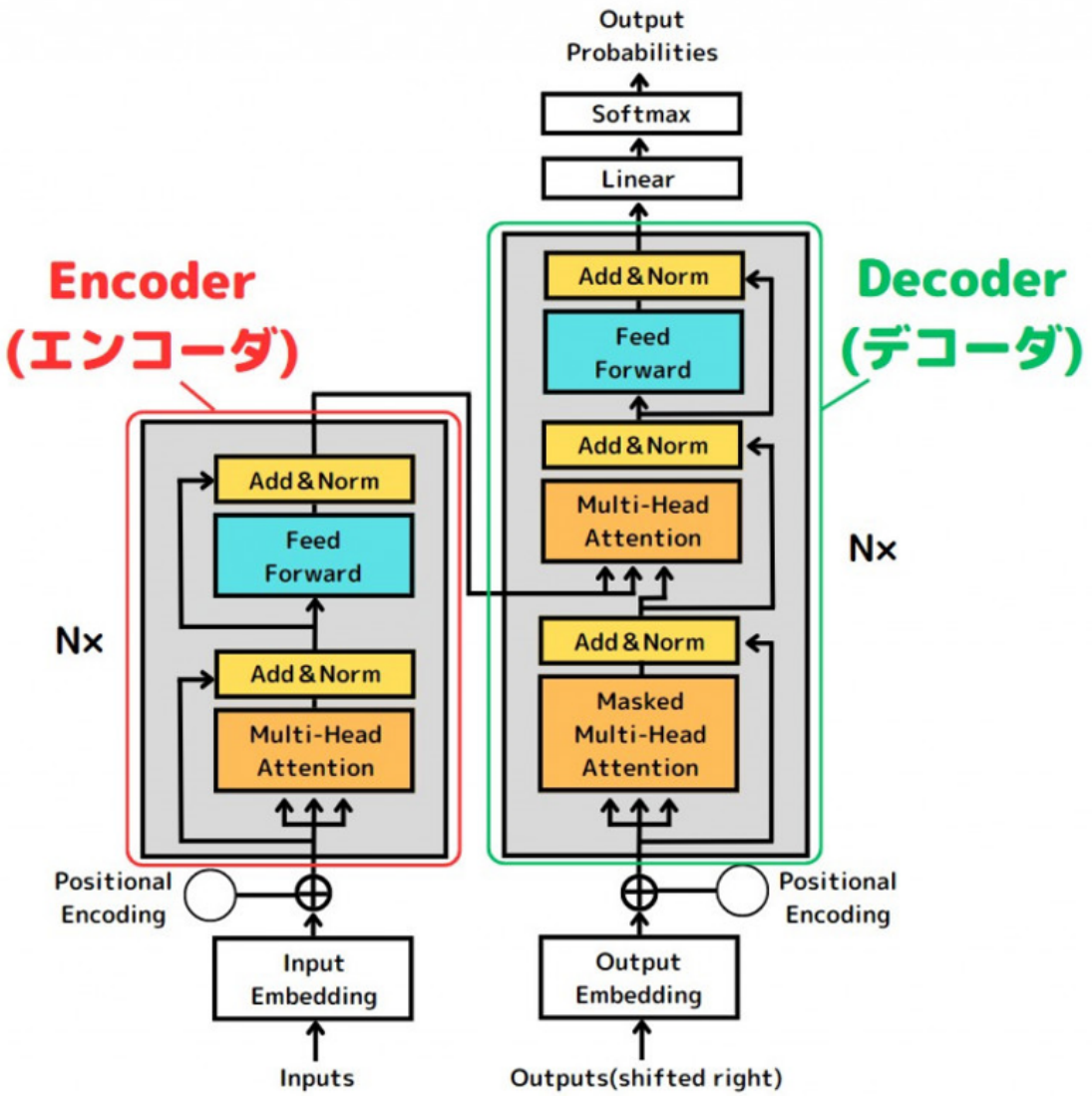

図 2.3: Transformer 結構

であり、デコーダーレイヤは、図 2.3 の右側の点線のボックスである。各エンコーダーレイヤ（またはデコーダーレイヤ）は、N 個のエンコーダー（またはデコーダー）を含み、基本的な Transformer モデルにおいて、N の値は通常 6 である。図から見て取れるように、入力のテキストシーケンスは、エンコードとデコードの 2 つの段階を経て、注意機構を組み合わせることにより、新しいテキスト特徴を持つシーケンスが生成される。

従来のシーケンスモデルである CNN や RNN と比較すると、Transformer モデルは、自己注意機構を導入することにより、単語間の関係を効果的に捉えることが可能となり、各入力単語のグローバルな情報をモデルに取り込むことにより、全体の性能を向上させる。さらに、このモデルは並列計算をサポートし、様々なハードウェア環境に適応可能で

ある。Transformer モデルは、その優れた特性から、自然言語処理の機械翻訳の分野で顕著な成果を収めており、他の分野でも広く応用されている。多くの NLP の事前トレーニングモデルの基盤として機能し、BERT モデルの前身として構築された Transformer は、現在に至るまで広く利用されている。

2.5　多音字認識の関連研究

中国語における文字から音素への変換（Grapheme-to-Phoneme、G2P）は、中国語のテキストを音声に変換するシステムの重要な構成要素である。この分野における最重要課題の一つは、多音字の発音における曖昧性の解消である。具体的には、文脈の中で与えられた多音字の正しい発音を決定することを目指している。先行研究によると、多音字の発音方法は主に、ルールベースと学習ベースの 2 つに分類される。

ルールベースの多音字認識方法 [5] [6] は、主に言語学の専門家が維持する強力な辞書と複雑な事前定義された規則に依存している。これらの方法は、テキストを単語の断片に分割し、辞書に一致する単語の断片の発音の曖昧さを取り除き、未確定の多音字の発音を決定するために手作りの規則を適用する。ルールベースの方法は多音字の認識において重要な役割を果たしているが、単語の断片が異なる意味を持つ場合、この方法は通常、正しい多音字の認識に難しさを抱える。例えば、「為」あるいは「為」などの同一漢字が異なる意味と発音を持つ場合がある。

一方で、学習ベースの方法は文脈情報を考慮して多音字の発音を決定する。これには、統計的なルール、決定木、最大エントロピー・モデル、ディープラーニングの手法が含まれる。学習ベースの方法においては、ディープラーニングの手法が多音字の発音から文脈特徴を抽出し、顕著な性能を発揮している。ディープラーニングに基づく多音字認識は、伝統的なディープラーニングモデルに基づくモデルと、事前トレーニングされた言語モデルに基づくモデルの 2 つに主に分かれる。

2.6　伝統的深層学習モデルに基づく手法

伝統的な深層学習手法において、中国語の多音字の曖昧性を解消するためには、条件付きニューラルネットワークと多階層埋め込み特徴に基づくアプローチが提案されている。論文 [7] では、文字ベクトル埋め込みと事前トレーニング済みの Word2Vec [8]

単語ベクトルテーブルを用いて、文字と単語を表現する。さらに、双方向長短期記憶（BiLSTM）[9] ネットワークを活用し、文章をエンコードして文章レベルのコンテキスト情報を取得する。このアプローチは、文字レベルと単語レベルの情報を組み合わせることで、多音字の曖昧性を減少させ、精度の向上を図る。BiLSTM でエンコードされた情報は、全結合層を経由し、最終的に softmax 層を通じて多音字の正しい発音を予測するために使用される。

論文 [10] では特定の多音字の近くのコンテキスト特徴を捉えるために、双方向長短期記憶（BiLSTM）層も使用されており、さらに、追加の品詞タグ（POS）と Word2Vec 埋め込みを利用して、文脈内でのより多くの情報を取得する。これにより、多音字の曖昧性をより効果的に解消し、精度の高い発音予測を実現する。

2.7　事前トレーニング言語モデル（PLM）に基づく手法

最近では、ゼロからトレーニングする予測モデルを代わりに、プレトレーニング言語モデル（PLM）が大量の未ラベルテキストデータで自己監督学習を行い、ファインチューニングを経てダウンストリームタスクに応用されるようになっている。

多くの研究により、PLM を使用することで、多音字の発音曖昧性解消問題の性能を改善できることが示されている。 例えば、論文 [11] は発音辞書と bert [12] を組み合わせて PDF のモデルを提案し、中国語のセグメンタルエラーとモデル構造の問題における従来のシステムの限界を解決した。PDF モデル入力は、各文字のエンコーディングに加えて、各単語のエンコーディングを加える。このアプローチにより、セグメンテーションエラーを回避し、発音辞書を効果的に利用することができる。 さらに、このモデルは、事前に訓練された言語モデル（PLM）をエンコーダとして使用し、入力シーケンスから文脈情報を抽出する。 これらの技術を組み合わせることで、多音節曖昧性解消タスクの精度と効率が向上する。

PDF モデル [11] の構造は、図 2.4 に示されている通りである。モデルはまず、入力文と発音辞書を照合し、すべての潜在的な語彙を取得する。これらの語彙はグリッド構造に構築され、後にトークンに平坦化される。この平坦化された構造には、入力文字とすべての一致した語彙が含まれる。このアプローチにより、モデルは文字とすべての単語セグメントを同時に処理し、分割エラーを回避することができる。また、多音字を含む潜在的な語彙に対しては、モデルが対応する音声特徴を提供する。

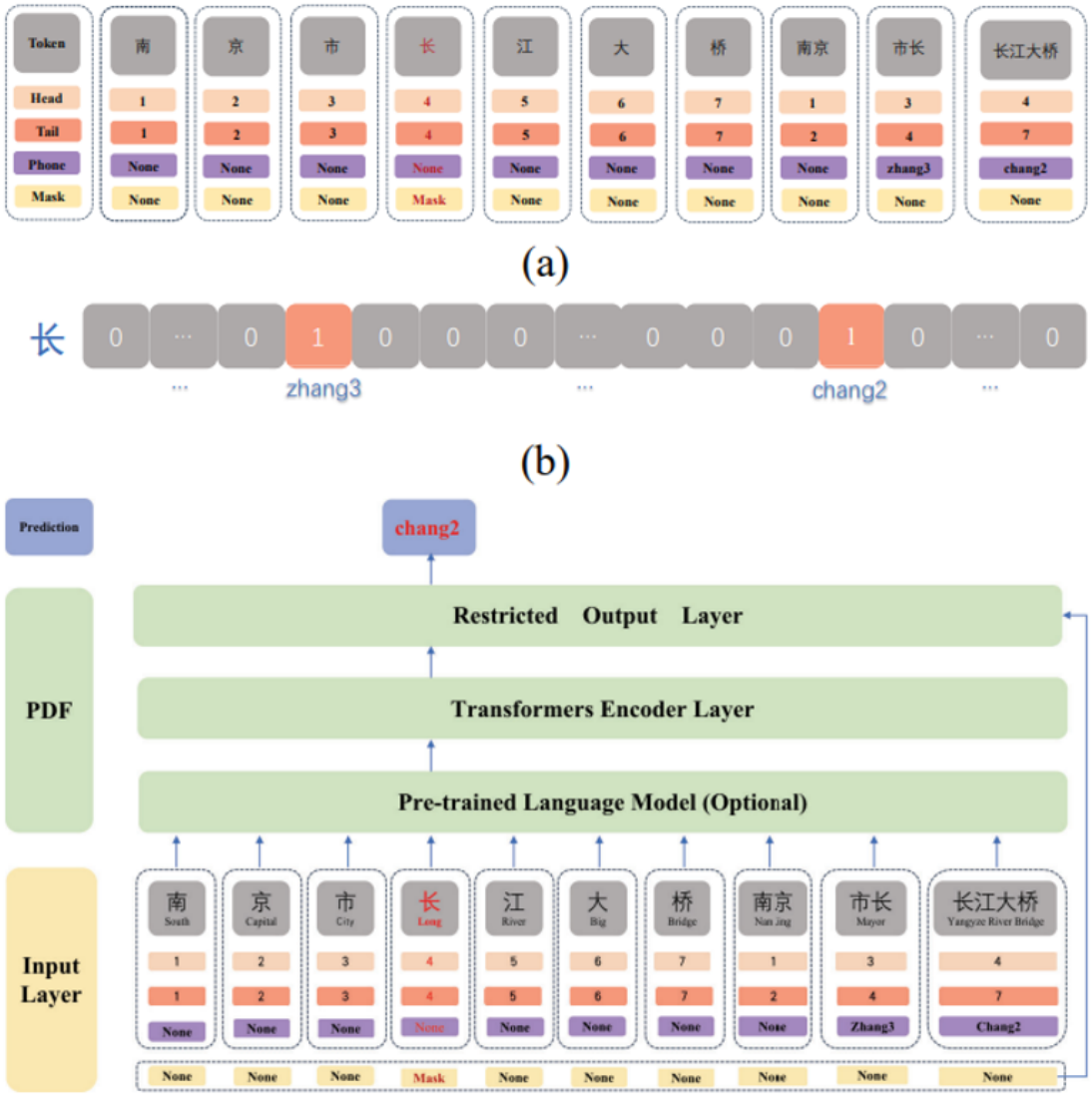

図 2.4: PDF 結構

しかしながら、この方法では入力情報に冗長な情報が含まれる可能性がある。これは、モデルの処理の複雑さを増加させる可能性がある。例えば、「南京」のようなトークンは、多音字の認識に直接寄与しない可能性がある。モデルの設計と最適化は、この複雑性に対処することを目指しているが、入力データの冗長性はモデルの効率と精度に影響を与える可能性がある。このため、モデルの構造とデータ処理アプローチの最適化は、多音字の認識精度を高めるために重要な役割を果たす。

g2pW [13] という多音字認識モデルにおいては、ターゲットの多音字およびその多音字の品詞タグ（POS タグ）に基づいて、学習可能なウェイトがついた softmax を使用し、BERT の出力を調整することで多音字の識別を実現する。g2pW の構築は図 2.5 を示す。

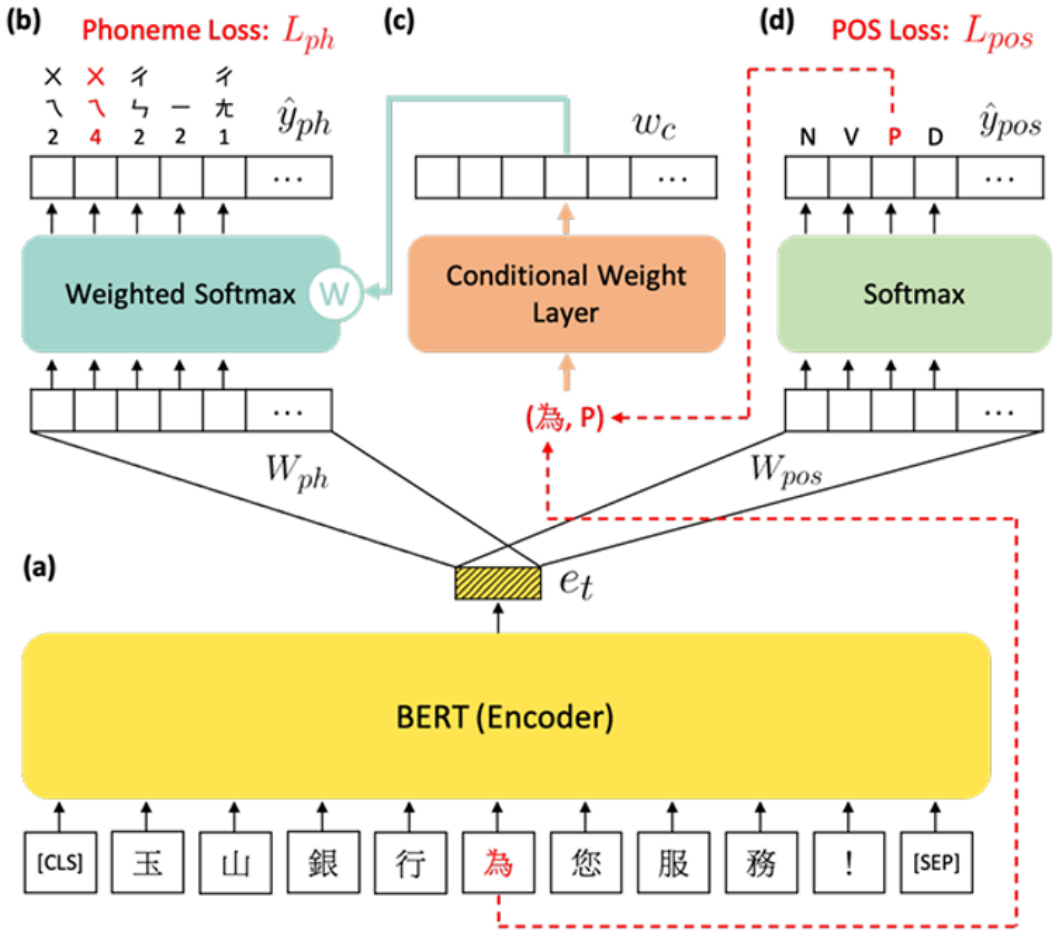

図 2.5: g2pW 多音字識別モデル結構

g2pW は、候補音素のための softmax 関数を学習する。具体的には、g2pW は、ターゲットの多音字およびその POS タグなどの補助的な特徴を利用して、softmax 関数内のウェイトを調整するための埋め込みを学習する。

同時に、このモデルは、多音字の発音の曖昧さと POS タグモデルをトレーニングするために、統一エンコーダ（BERT）を使用する。さらに、g2pW は、conditional weight layer の入力として、共同トレーニングされたラベルモデルの POS タグの予測を使用する。このモデルは PDF と比較して入力が簡素化されるが、モデルのタスクはより複雑になり、多音字の発音のタスクに加えて、単語の特性を予測するタスクも追加される。ただし、一部の多音字は、異なる単語で異なる品詞を持つ場合がある。例えば、「行走」では多音字「行」は動詞として使用され、「歩く」を意味するが、「銀行」では名詞として使用され、「金融機関」を意味する。

そのため、多音字の品詞は異なる単語で異なる場合がある。しかし、モデル g2pW は

トークンとして個々の文字に依存しており、単語全体ではないため、文中で構成される単語の品詞を予測する際には課題が発生する可能性がある。これは、モデルが多音字の使用文脈を細かく理解し、予測する必要があり、このプロセスがモデルのトレーニングを難しくする可能性があるためである。

第 3 章

BERT モデル

Transformer は NLP 領域において重大な変革をもたらしたが、文脈に関連する意味情報を真に理解するためには、研究者たちは迅速にさらなるモデルの必要性を認識した。この背景から、BERT [12]（Bidirectional Encoder Representations from Transformers）が 2018 年に登場し、自然言語処理領域における重要な進展となった。BERT の事前トレーニング言語モデルは、Devlin らにより提案された Transformer に基づく双方向エンコーダ表現モデルであり、単語に対して動的な表現が可能である。BERT モデルは、埋め込み層、Transformer エンコーダ層、出力層から主に構成され、その構造は以下の図 3.1 に示されている。

BERT モデルは、Transformer 層のエンコーダー部分を主体として構築され、各単語の特徴を抽出する際、文脈の内容が各単語に与える影響を十分に考慮する。これにより、モデルは強力な言語表現と特徴抽出能力を有するようになる。BERT-BASE モデルは 12 の Transformer 層、768 次元の隠れ層次元、12 の注意機構を有し、総パラメータ数は 1.09 億に達する。また、BERT-LARGE モデルはさらに膨大な 3.4 億のパラメータを持つ。これらの膨大なパラメータ数により、BERT モデルの推論速度は遅く、トレーニングおよびデプロイの際のハードウェア環境に高い要求が生じる。

3.1 前処理

ネットワークモデルは、テキストを直接入力することができず、データセットのテキストを事前処理する必要がある。これにより、テキスト内の文字、単語、句読点を数字のマークに変換することができる。BERT モデルは、WordPiece アルゴリズムを使用し

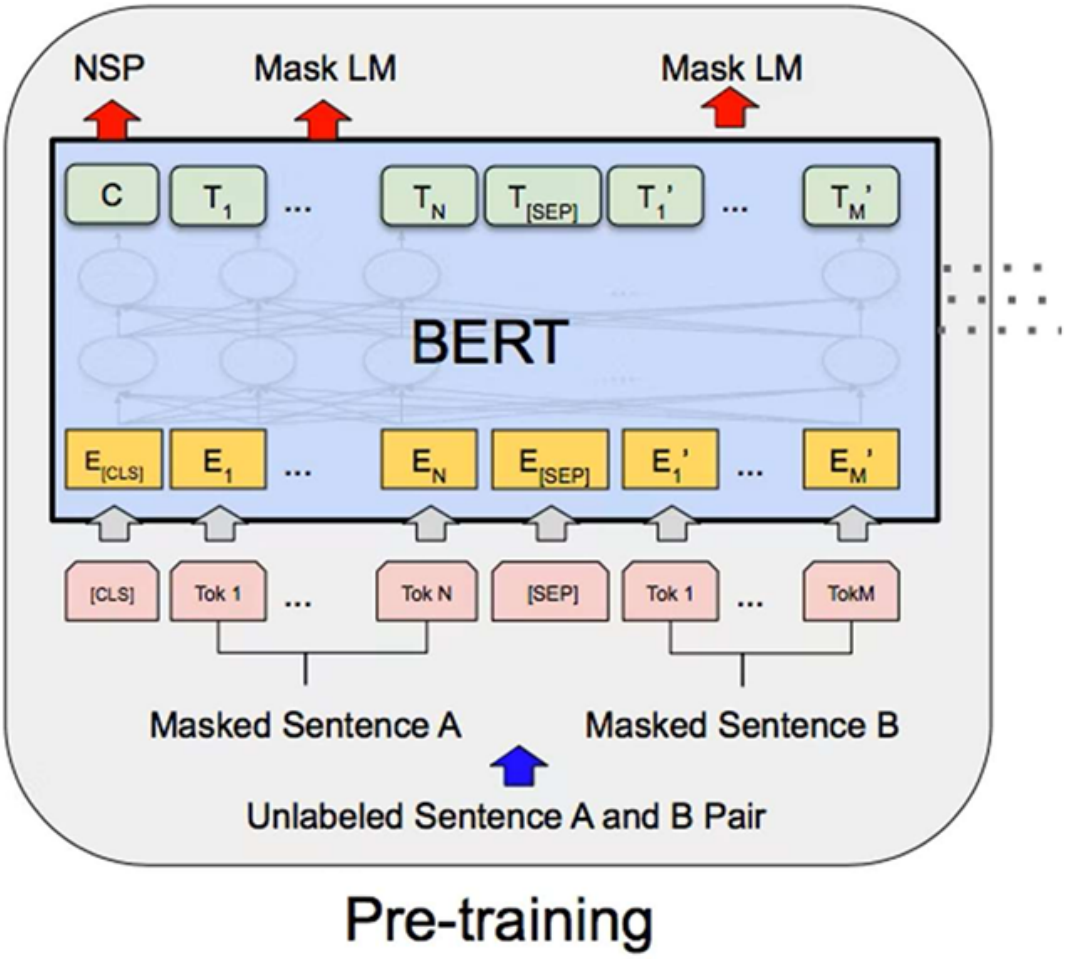

図 3.1: Bert モデル

て語彙を構築し、生成された語彙に基づいて入力テキストを対応するトークン列に変換する。語彙表に存在する単語は、事前処理時に直接対応する数字に変換される。例えば、「hello」という単語は直接数字 7592 に変換される。一方で、語彙表に存在しない単語は、処理時に分割され、その後分割された部分が変換される。例えば、「it's」は「it」、「'」、「s」に分割され、最終的に (2009、 1005、 1055) に変換される。以下の図 3.2 は、テキストがトークン列に変換される例を示している。

マーク列に変換された後、シーケンスの先頭と末尾に特殊なマーク「[CLS]」（テキストの開始を示し、分類タスクに使用）と「[SEP]」（テキストの終了を示す）が追加され、「[CLS] 文 [SEP]」の形式が形成される。2 つのテキスト入力がある場合、前処理では生成された 2 つのマーク列を連結し、「[CLS] 文 A[SEP] 文 B[SEP]」の形式が形成される。生成されたシーケンスの長さがモデルの入力長より短い場合、マーク列の末尾に「[PAD]」

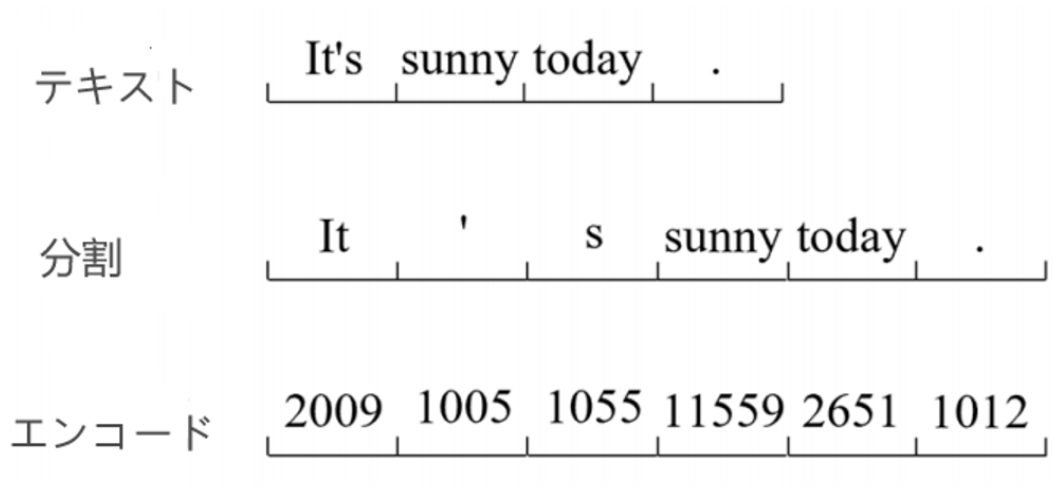

図 3.2: テキストを数字のマーク列に変換する概略図

マークを追加して入力長の要件に合わせる必要がある。シーケンスの長さが入力長より長い場合、必要に応じて切り詰めるか、複数の入力に分割することができる。

3.2　埋め込み層

BERT モデルの埋め込み層には、単語埋め込み、位置埋め込み、文埋め込みが含まれる。トレーニングサンプルは前処理を経て埋め込み層に入力され、対応する計算を経て、三つの異なる特徴ベクトルが得られる。これらのベクトルを合算し、埋め込み層の出力となる特徴ベクトルが得られる。具体的なプロセスは図 3.3 に示されている。

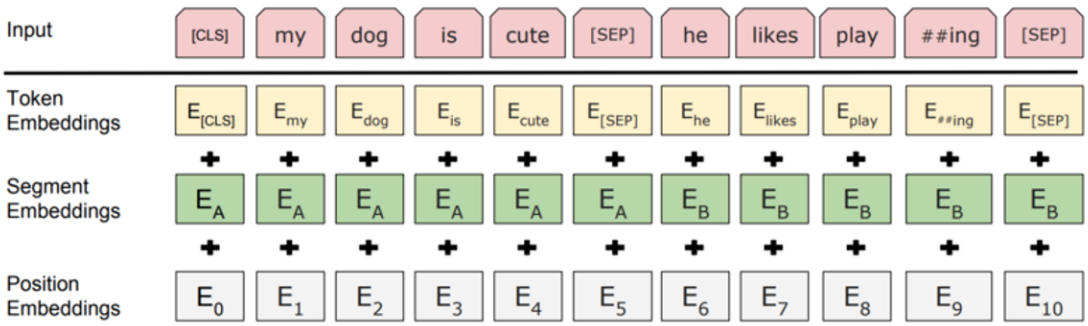

図 3.3: 単語埋め込みの結構図

過学習の問題を緩和し、データ分布を最適化するために、モデルは埋め込み層の出力前に Dropout とレイヤー正規化を行う。Dropout は、指定された確率でランダムに一部の特徴を破棄することにより、トレーニングプロセス中に異なるネットワーク構造を生成

し、モデルの汎化能力を向上させる。レイヤー正規化は、出力データを標準正規分布に正規化することで、データを安定化させ、トレーニングプロセス中のパラメータの変化による内部共変量シフトの問題を防ぐ。

3.3　Transformer 層

Transformer 層は BERT モデルの中核を成し、複数のエンコーダー側の Transformer ブロックが積み重ねられている。各 Transformer ブロックは、複数のヘッドを持つ注意機構とフィードフォワードネットワークを含んでおり、中間の出力データも Dropout とレイヤー正規化を経て、データの安定性が保たれる。基本的な構造は以下の図 3.4 に示されている。

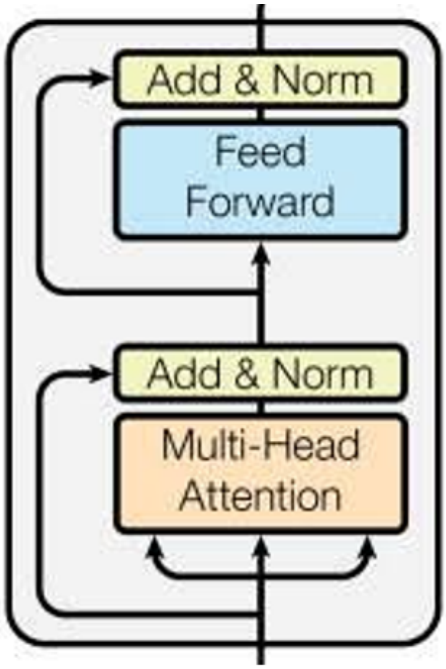

図 3.4: 単語埋め込みの結構図

注意機構は Transformer の中で特殊な構造を持ち、「query-key-value」の三つの行列（WQ、WK、WV）で構成される。これは異なる位置の相互作用を決定し、それによって出力データを得るために使用される。入力シーケンス X に対して、まず三つの行列との行列積演算を行い、対応するクエリ行列 Q、キー行列 K、およびバリュー行列 V を得る。ここで、$Q, K, V \in \mathbb{R}^{m \times d_k}$ あり、m は入力シーケンスの長さ、d_k は注意機構の行列の次元を示されている。注意機構の計算過程は、以下の式に示されている。

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{3.1}$$

独個の位置に対する全ての位置の影響を確定するため、注意機構は Q 行列の第 i ベクトルと K 行列の全てのベクトルのドット積を行い、第 i 位置への影響の評価スコアを得る。これは、Q 行列と K 行列の転置の行列積により迅速に計算され、各位置間の影響の評価スコアが得られる。次に、このスコアを d_k で割り、勾配の安定を図る。そして、Softmax 関数を用いて各位置間の影響の割合を計算し、これにより注意行列が得られる。計算の手順は以下の式に示される。

$$\text{Softmax}([z_1, z_2, \ldots, z_n]) = [q_1, q_2, \ldots, q_n] \tag{3.2}$$

データの前処理中に短いテキストに対してパディング処理が行われるため、埋め込み位置が結果に影響を与えないようにする必要がある。そのため、注意機構の計算前に、前処理中に生成された内容シーケンスに基づいてパディング位置のスコアを負の無限大に設定する。これは、limx →∞ ex=0 関数によって計算されると、対応する位置の値が 0 になり、結果にパディング処理が影響を与えないようにするためである。最後に、得られた注意機構行列と V 行列との行列積を取り、入力データの特徴抽出が完了し、より高次元のデータ特徴表現が得られる。

注意機構の計算過程から分かるように、注意機構行列は各位置と前後の全ての位置との影響の度合いに基づいており、その結果は双方向の特徴表現となる。マルチヘッド注意機構は、複数の注意機構の出力を結合し、線形層 W を使用して対応する出力次元に変換することで、Transformer 内の注意点を増やすことができ、ネットワークがより豊富な文脈特徴を抽出できる。その構造は以下の図 3.5 に示されている。

もう一つのフィードフォワードネットワークは、2 つの線形層と Gelu 活性化関数から構成され、簡単な「線形層-活性化関数-線形層」構造を形成する。マルチヘッド注意機構の出力は、Dropout とレイヤー正規化の処理を経て、1 つの線形層を使用して結果をより高次元の特徴空間にマッピングし、Gelu 活性化関数によって処理された後、別の線形層でネットワークの隠れ層の出力次元に変換される。

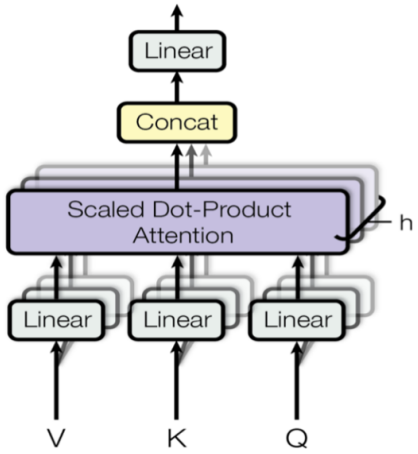

図 3.5: マルチヘッド注意機構の構造図

3.4　事前学習タスク

BERT モデルには 2 つの事前学習タスクがある：マスク言語モデリングタスクと次の文予測タスクである。マスク言語モデリングは、モデルを特定のコーパスに適応させるタスクであり、BERT モデルの事前学習には Wikipedia および BookCorpus という 2 つの大規模な英文コーパスが使用された。マスク言語モデリングは自己教師学習に属し、補助関数を使用してコーパステキストから自己監督情報を抽出する。その後、抽出された自己監督情報をネットワークのトレーニングに使用し、モデルがテキストの特徴表現を学習できるようにする。

補助関数は、前処理されたトークン列にランダムにマスクをかける。特別なマーク以外の位置のトークンは、15% の確率でマスクされ、選択されたトークンには次の 3 つの処理方法がある：

1. 80 %の確率で [MASK] という特殊なトークンに置き換えられる。例えば、「This is a house.」は「This is a [MASK].」に置き換えられる。
2. 10 %の確率でランダムなトークンに置き換えられる。例えば、「This is a house.」は「This is a tree.」に置き換えられる可能性がある。
3. 10 %の確率で変更されない。

BERT モデルは、マスク処理されたテキストを入力として使用し、選択された位置の元のトークンをラベルとしてトレーニングする。モデルは Transformer 層のマルチヘッ

ド注意機構を利用して、マスクされた位置のコンテキストを分析し、マスクされた位置の元のトークンを予測する。

自然言語処理のタスクには推論タスクが含まれており、これらの推論タスクは 2 つの文の関係を分析してモデリングされる。次の文予測タスクでは、コーパスから文 A と文 B をトレーニングサンプルとして選択し、50 %の確率で文 B が文 A の実際の次の文である場合に「IsNext」のラベルが付けられる。残りの 50 %の確率で文 B がコーパスからランダムに選択された文である場合には、「NotNext」のラベルが付けられる。したがって、BERT モデルは次の文予測タスクを通じてモデルをトレーニングし、2 つの文の間の関係を理解できるようになる。

ネットワークモデルは事前トレーニング後、トレーニングデータからテキストの一般的な表現を学習する。ファインチューニングでは、下流のタスクに基づいてモデルの出力層の構造を調整し、小さな学習率でモデルをトレーニングする。事前トレーニングモデルを使用することで、ネットワークをゼロからトレーニングする必要がなくなり、モデルの学習効率が向上する。

第 4 章

提案手法

本章では、初めに本研究の目標と研究のアイデアについて述べ、次に、実験の前処理、実験のモデル構造、入力と出力の紹介の順に、実験のデザインに関する考え方について説明する。

4.1 デザインのアプローチ

多音字の発音は通常、文脈によって決定されるため、現在広く使用されている意味理解モデルと比較して、BERT モデルは双方向の事前トレーニング方式を利用し、テキスト理解能力に優れている。そのため、多音字の音読み認識タスクには事前トレーニング済みの BERT モデルを利用することとする。

本実験は、対象となる予測が文中の各多音字の読み方であるため、単語の読み方と単語の意味は同じ概念ではない。ほとんどの漢字の発音は、その単語と一定の関連性があるが、例えば「和面」、「和気」のような単語では、「和」の発音がそれぞれ「huo」、「he」と異なる。単語を知っていれば字の発音を判別できるが、中には単語だけでは発音を確定できない多音字も存在する。中国語の文脈の複雑さにより、所属する単語だけでは発音を一意に判断できない多音字も多く存在する。例えば、「地」や「的」といった漢字は、しばしば文章の接続部に単独で現れ、文脈によって発音が異なる。このように、本実験の予測対象である発音は、その多音字が属する単語だけでなく、文の文脈とも一定の関連性がある。

したがって、Bert モデルの出力の各単語の埋め込みとして単純に使用すると、その単語内の多音字の発音を正確に判断するのは難しい。そこで、各字を単語ではなく個別に

トレーニングモデルの入力とする提案を行う。これにより、通常はラベル（POS）に頼る伝統的な発音選択プロセスが簡略化される。通常のランダムマスクタスクやテキスト分類の代わりに、トレーニングデータのテキストに多音字の発音を追加してトレーニングを行う。この方法により、self-attention などの双方向相互作用のトレーニング中に、発音と文中の各字の情報が交換され、統合される。さらに、モデルがテキストを効率的に理解し、トレーニングタスクをより効果的に達成できるように、発音部分に対しては通常のランダムマスク処理ではなく、指向マスク処理を行う。これにより、モデルは発音と全体の文の関係をより良く理解できるようになる。

そして最終的なテストプロセスでは、予測された発音を出力する。このため、字をトークンとしてモデルに取り入れ、方向性のある発音マスクのトレーニングタスクを定める方法を提案し、モデルが字同士や字と発音の関係をより良く理解し、多音字の識別タスクを効果的に遂行できるようにする。

さらに、一部の多音字が属する単語の品詞とその発音には一定の関連性があるため、多くの研究者は多音字の品詞も入力として使用している。本論文では、この考え方に触発されて、多音字の発音入力を持つ新しい多音字認識モデルを設計した。総合的には、本論文では 2 つの新しい多音字認識方法を提案している。一つは多音字の発音入力を備えた多音字認識方法であり、もう一つは多音字の発音とその多音字が属する単語の品詞の両方を入力とする多音字認識方法である。

4.2　データ処理

上記のアプローチに基づいて、本実験では以下の具体的な手順を設計した。まず、意味、文脈、品詞が多音字の発音に影響を与えることを考慮し、複雑な問題を簡略化することを目指す。そのため、トレーニング段階で、多音字の発音を直接文の中に挿入することにした。これにより、モデルは発音の位置に基づいて、その発音が指す文字の発音を直接判断できる。

同時に、モデルが入力された発音のフォーマットを認識するために、既に訓練された中国語識別モデル bert-base-chinese を使用する。これは大量の中国語テキストでトレーニングされ、各文字の発音と漢字を識別するために使用できる。また、文字が多音字であるかどうかは既知であり、その判断は容易である。巡回を行うことで、文の中の多音字の位置を確認することができる。トレーニングデータは、正しい発音を持つ大量の文で構成

される。一文が一つのトレーニングテキストとなるように、図 4.2 に示されている。

次に、文を処理する際には、既知の多音字セットに基づいて文内の多音字を選別し、その多音字の正しい発音を挿入する。モデルのトレーニングを簡素化し、難易度を下げるために、各テキストでは一つの多音字のみを対象とする。一文に複数の多音字が存在する場合は、複数回のトレーニング認識が行われる。同じ文の処理対象の多音字が異なる場合、それらは異なるデータに分類される。その後、図 4.1、4.2 に示されているプロセスを通じて、データの前処理を行う予定である。

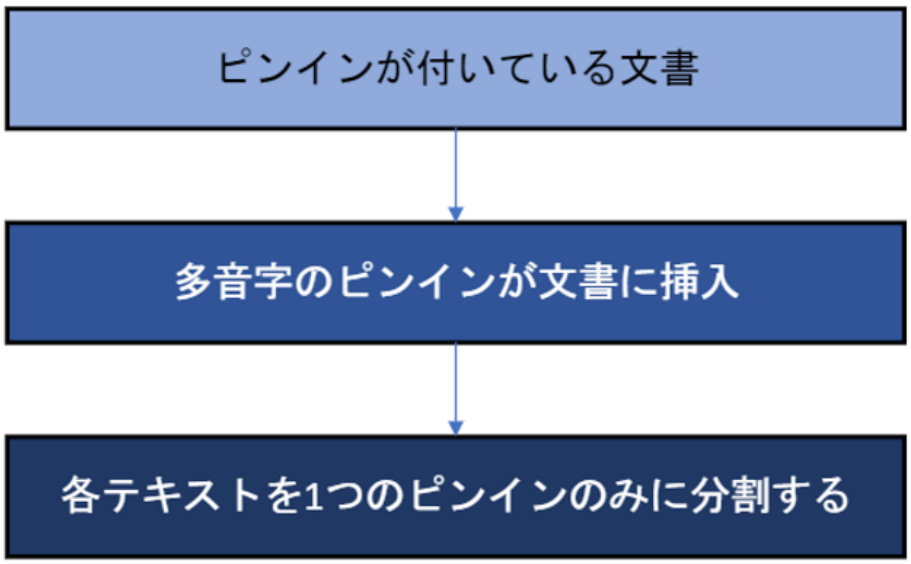

図 4.1: データの前処理プロセス

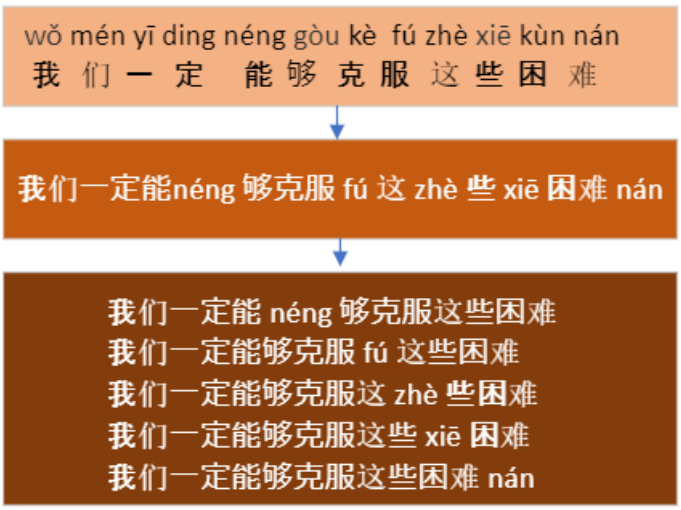

図 4.2: データ前処理の例

多音字の発音と単語の品詞を用いて認識手法

この方法の入力は、第一の手法における前処理後の入力に、対象の多音字が含まれる単語の品詞を該当する多音字の後ろに追加することである。この手法においては、まず、品詞を識別することが可能な前処理済みの BERT を用いて、元の文をトークンに分割する。その後、全てのトークンに対して品詞の識別を実施する。最終的に、対象となる多音字の品詞を抽出し、第一の手法の前処理結果に加える。具体的な手順は図 4.3 に示されている。

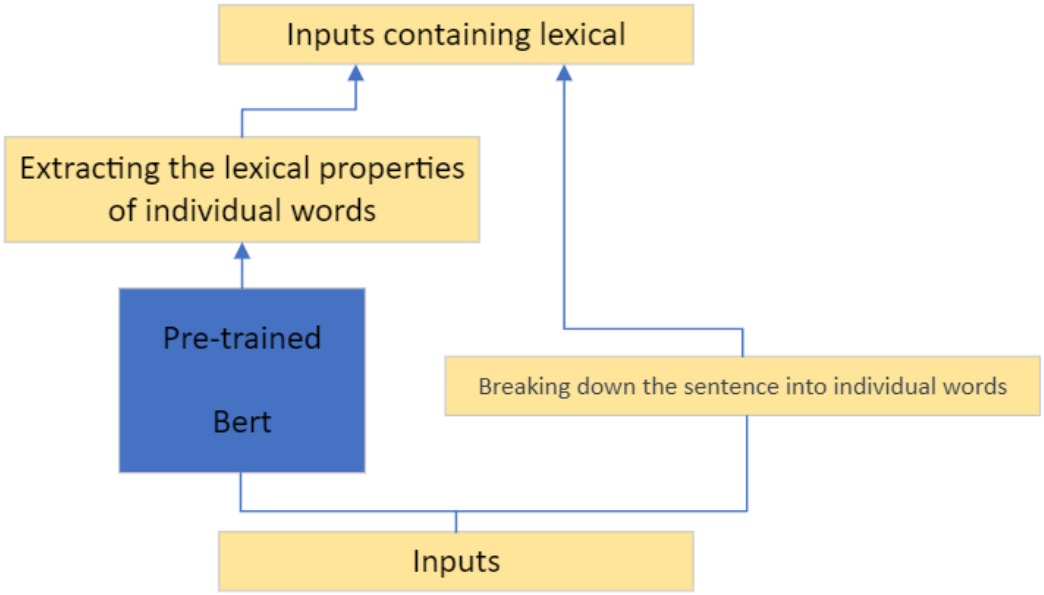

図 4.3: 品詞を用いて手法プロセス

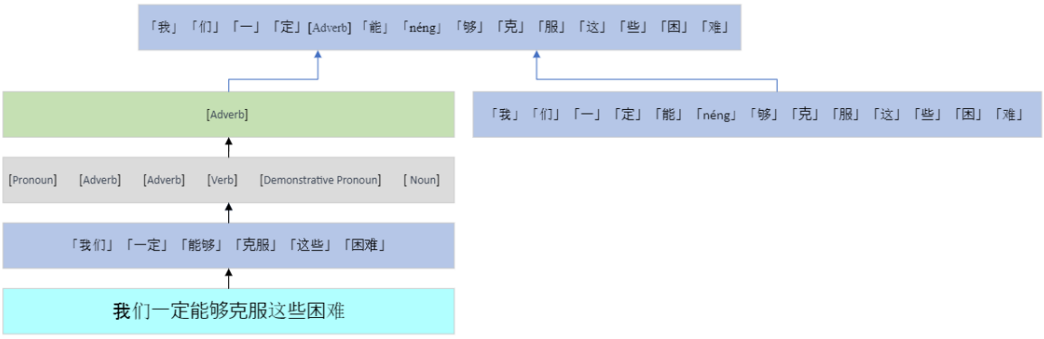

図 4.4: 前処理プロセス

4.3　モデル構造

通常の場合、BERT モデルは膨大なモデルサイズ、すなわち数十億のパラメータを有しているため、これはトレーニング時間とサーバーメモリにとって大きな挑戦となる。そこで、本論文では BERT モデルを簡素化することを目指し、モデルのトレーニング効率を向上させ、サーバーメモリの負担を軽減することを望んでいる。

そのため、本研究では伝統的な BERT モデルを簡略化し、BERT の中核層と構造を直接採用し、self-attention と feed forward から構成される attention layers を構築した。これは transformer のエンコーダに似ており、具体的な構造は図 4.5 に示されている。通常の BERT は少なくとも 12 層の transformer で構成されているが、本論文では、各字をトークンとして使った方向性の mask が多音字認識タスクを効果的に実現できるかどうかを検討している。そのため、小さなネットワーク構造を設計し、モデルを容易にトレーニングできるようにしている。ただし、小さすぎるネットワークモデルは情報の抽出能力を低下させる可能性があるため、本論文では 6 層の attention layers から成る多音字認識モデルを構築した（図 4.5)。

位置エンコーディングは文の順序情報をモデルに効果的に伝えることができ、非常に重要な処理手順である。多音字認識のタスクは分類タスクの一環であり、本論文ではモデルの最後の Linear を全結合層として設定し、前段階で抽出された全文の特徴を最終的な出力空間にマップする。この出力空間はクラスラベルであり、本論文の多音字認識タスクでは、クラスのラベルはすべての多音字の発音である。最後に SoftMax 活性化関数を介してクラスの確率分布を生成し、ネットワークの最後の層の出力を各クラスの確率を示す分布に変換する。各予測結果とその予測結果の予測率が出力される。推論時には、各クラスの確率値に基づいて、モデルの最終的な予測結果、つまり入力テキストの多音字の発音を選択することができる。

4.4　モデルの入力と推論

モデルの入力に際しては、先に述べたデータの前処理に加え、トークン化、整数識別子への変換、ゼロ埋め、発音のマスキングなどのステップが必要である。入力の例を図 4.6 に示す。

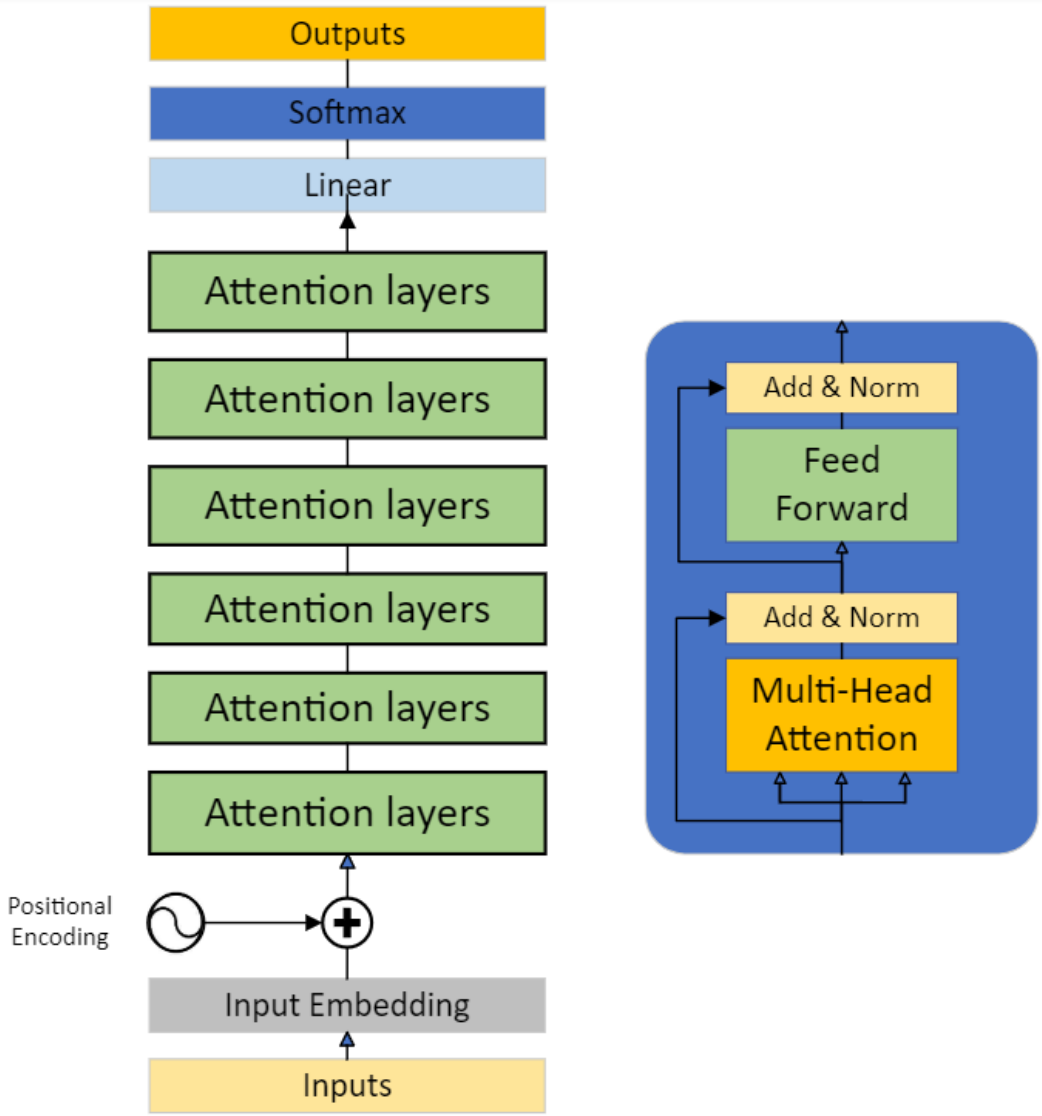

図 4.5: 多音字モデルの構築（左） と attention layers の構築（右）

図 4.6 のケースは、多音字の発音を含む多音字認識手法の一例であるが、同時に多音字の発音と該当多音字が属する単語の品詞を含む入力における多音字認識手法においては、最下層の入力テキストを品詞情報を含む形に変更する。

前処理データには読み方の真値が含まれているため、これをそのままモデルのトレーニングデータとして使用すると情報漏洩が発生する可能性がある。そのため、発音のトークン化時に、モデルはランダムな数字を発音の整数識別子として選択する。そして、テキストの全ての整数識別子を高次元空間に投入し、モデルに供給する。モデルの入力データには、文の全ての整数識別子だけでなく、発音の位置とゼロ埋めの位置情報も含まれている。

ゼロ埋めの位置情報は、通常の BERT モデルと同様に機能する。自然言語処理におい

ては、シーケンスを同一の長さに調整し、バッチ処理やモデルへの入力を容易にするためにゼロ埋め（Padding）が一般的に使用される。BERT モデルでは、ゼロ埋めの位置情報が入力シーケンスを固定長にするために利用される。この種のパディングは、小規模なデータセットでのトレーニングにおいて重要である。なぜなら、ニューラルネットワークは並列計算を行うために固定長の入力を必要とするからである。例えば、一つのバッチが 5、7、4 個のトークンを含む 3 つの文を含む場合、これらの文をモデルに同時に入力するためには、最長の文の長さに合わせてこれらの文を同一の長さにパディングすることが一般的である。BERT モデルは自己注意機構を用いて入力シーケンス内の全てのトークンの関係を考慮する。ゼロ埋めの位置では、BERT モデルはパディングされたトークンを無視し、それに対する計算を行わない。これは、モデルの処理中にパディングされたトークンが実際のシーケンスの表現に影響を与えないことを意味する。

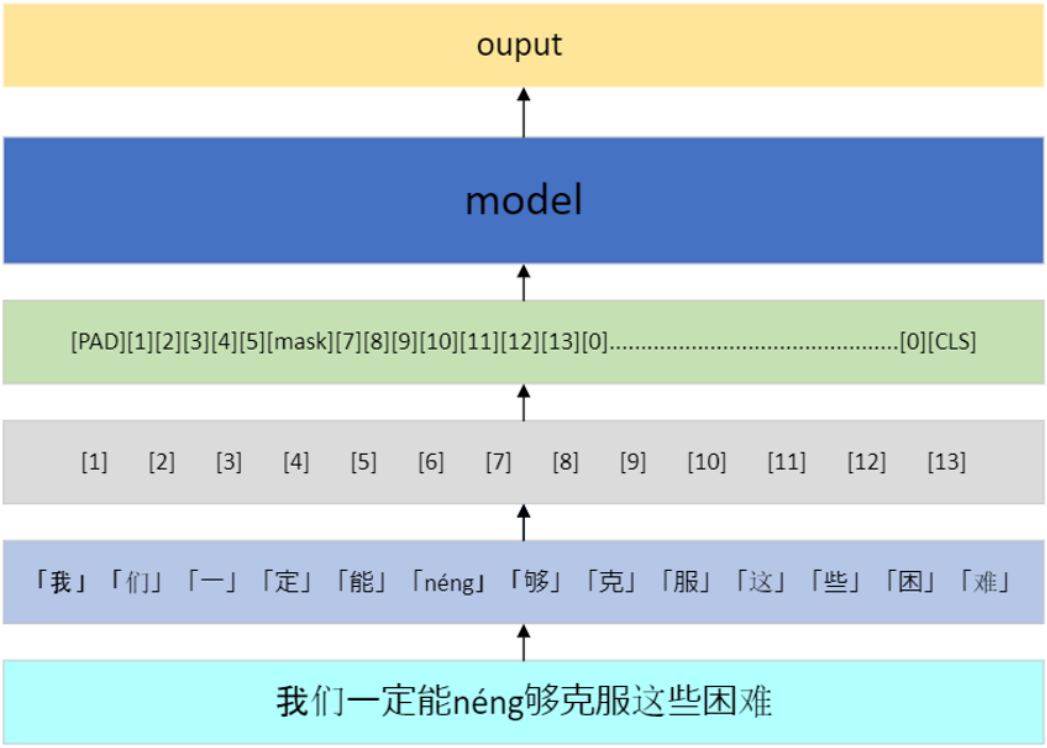

図 4.6: 入力のケース

また、発音の位置はマスクとして設定され、モデルは発音を遮蔽し、通常は発音の整数識別子を導き出すためにトレーニングされる。モデルの出力前には、全結合層と SoftMax 処理が行われる。全結合層によって、1 × 30000 のベクトルが生成される。トークンの数が 20000 を超えるため、本研究では全結合層の出力を 30000 と設定した。当然ながら、

埋め込み処理時のモデルパラメータも 30000 である。全結合層と SoftMax の出力は、その発音が 30000 のトークンの各トークンに対してどの程度の可能性を持つかを示すベクトルである。もしモデルが正確に発音を識別できれば、最終的な 1 × 30000 のベクトル内で最大値を持つトークンが正しい発音となるはずである。テスト段階では、多音字の前に特殊な文字を追加し、モデルがこの文字に遭遇した際に、自動的にその文字をマスクする。最終的なモデルの出力は、モデルが推論した多音字の発音である。さらに、各多音字の読みは固定されているため、本研究では最終的な出力結果を得た後、SoftMax 出力の最大値の位置に対応するラベルを直接使用するのではなく、まず、入力された多音字に対応する固定された読みの SoftMax 出力を抽出し、その大きさを比較して最も可能性の高いものを選択する。この手順は図 4.7 に示されており、この図では多音字の入力が A、B、C の 3 つの読み方があると仮定されている。これにより、SoftMax 値が最大であっても多音字の読み方ではないという誤りを避けることができる。

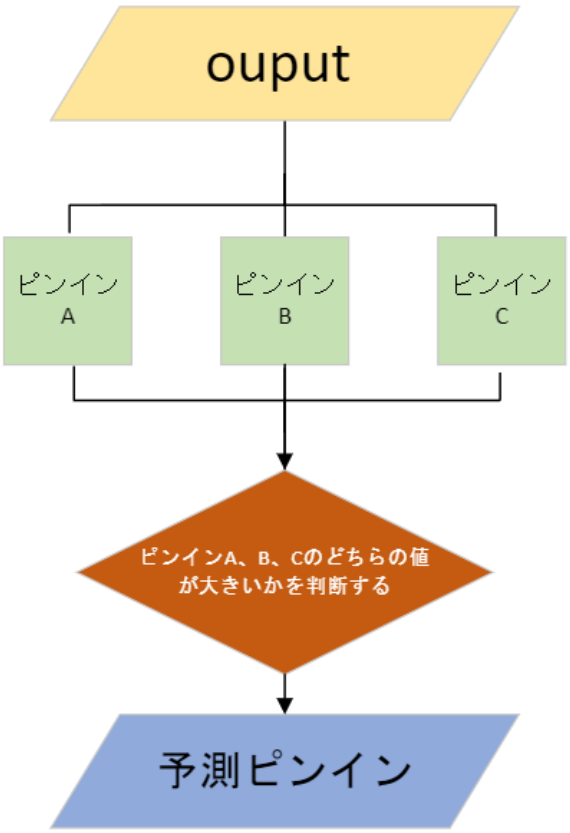

図 4.7: 予測プロセス

第 5 章

実験

第 5 章では、実験の設計を紹介し、実験設定（第 2 節）、データセットの収集、データの前処理、実験結果（第 3 節）、および追加実験にわたる内容を総合的に述べる。

5.1　データセット

本研究では、多音字のトレーニングデータとして、多音字のセットデータと正しい発音が付された大量のテキストデータを含む。

多音字のセットには、すべての多音字と各多音字に固有の複数の発音が含まれている。表 1 には、多音字データの一部例を示している。多音字セットの総文字数は 3226 文字である。各多音字の発音数は異なる可能性があり、大多数の多音字は 2 つの発音のみを持つが、最多の発音数は 6 つであり、異なる発音数の統計は表 2 に示されている。

第 4 章のデータ前処理では、多音字データのみを抽出する必要があり、この際に多音字が多音字セットに含まれているかどうかで多音字であるかを判断する。分詞段階では、中国語テキストに特化した事前にトレーニングされた BertTokenizer のインスタンスを使用する。BertTokenizer は BERT モデルに関連するテキスト処理ツールであり、元のテキストをより小さな単位（「トークン」と呼ばれる）に分割する。本論文では、PyTorch フレームワークの「bert-base-chinese」という名前の BertTokenizer のインスタンスを使用する。この BertTokenizer のインスタンスは、テキストの各漢字や記号、または引用文の単語を個別のトークンに分割できる。しかし、BertTokenizer を直接使用すると、一部の発音が複数のトークンに分割される可能性がある。例えば、「的」という漢字の一つの発音「de」は、「d」と「e」に分割されることがある。そのため、本論文では、まず、

多音字のすべての発音を一括して BertTokenizer の辞書に追加し、BertTokenizer が発音を正確なトークンに分割できるようにする。

表 5.1: 多音字データの例

多音字	各多音字の発音			
的	dí	dì	de	
了	liǎo	le		
有	yǒu	yòu		
這	zhè	zhèi		
个	gè	gě		
上	shàng	shǎng		
着	zhuó	zháo	zhāo	zhe
得	dé	děi	de	

表 5.2: 多音字の発音統計

総多音字個数	3226				
発音数	2	3	4	5	6
多音字個数	2629	508	64	24	1

本研究においては、総計 460,547 件のテキストデータを利用した。データは訓練データ、検証データ、テストデータとして、比率を 6:2:2 に分割した。データセット内において、一部の多音字が一つの発音のみを持つ状況や、全ての発音が表れていない場合があるため、これがモデルのトレーニング及び精度に与える影響を考慮し、多音字の発音の出現状況を統計することが重要である。このため、本論文では多音字の組み合わせの尺度を定義した。その値は、各多音字とそれに対応する固定された発音の組み合わせの数を示す。例えば、多音字「的」には ’dí’、’dì’、’de’ という三つの発音があり、一つの多音字と一つの発音は一つの多音字の組み合わせとなる。したがって、「的」には ’ 的’ ’dí’、’ 的’ ’dì’、’ 的’ ’de’ の三つの多音字の組み合わせが存在する。

各データセット内で多音字の発音がどのように現れているかを説明するために、多音字の組み合わせの数量を使用した。各データセットの詳細な情報は表 3 に示されている。

さらに、異なるデータセットで同じおよび異なる多音字及び多音字の組み合わせがどのように影響するかも検証するために、本研究では異なるデータセット間での多音字と多音字の組み合わせの交差及び結合の状況も統計し、その詳細は図 5.1、図 5.2 に示されている。

表 5.3: 各データセットの統計

	テキスト	多音字	多音字組
Train	276328	667	913
Validation	92110	587	763
Test	92109	579	744

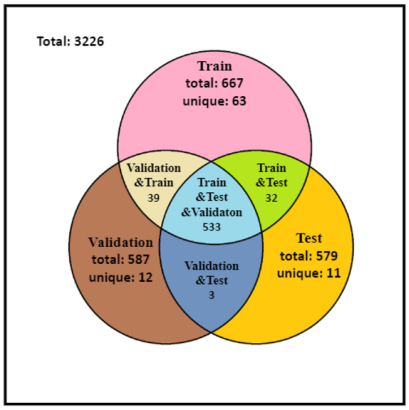

図 5.1: 各データセットの多音字の統計

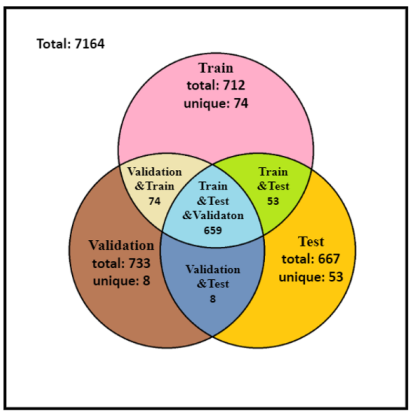

図 5.2: 各データセットの多音字組の統計

5.2　実験設定

本研究では、最適化関数として Adam を採用した。トレーニング段階での学習率の適切な調整のため、逐次的な学習率減衰方法を採用することで、モデルは損失関数の最小値をより精確に見つけ、性能を向上させることができる。さらに、小さな学習率はモデルがトレーニングデータに過剰に適合するリスクを減少させ、未知のデータに対するモデルの汎化能力を向上させるのに寄与する。また、トレーニングの初期段階で、モデルが最適解からまだ遠い可能性があるため、大きな学習率を使用して素早く進み、トレーニングの収束速度を向上させる。実験の詳細は表 5.4 に示されている。

表 5.4: 実験詳細

ハイパーパラメータ	値
Vocabulary Size	30000
Learning rate	0.00001
Batch size	128
Heads multi-Head attention	12
K Q V mebedding size multi-Head attention	64
Embedding Size	768
Text max size	60
FeedForward hidden layer Dimension	3072
Epoch	30

また、本論文では、単語の品詞を識別するための事前学習モデルとして fastHan モデルを採用している。このモデルは、BERT をベースにした多目的モデルであり、中文分詞、品詞識別などの複数の機能を備えている。13 のコーパスでトレーニング及び評価され、品詞識別のタスクで優れた性能を発揮し、最新の技術水準に近い、あるいはそれに達している。fastHan は、非トレーニングコーパスでも優れた性能を示し、モデルの強力な転移及び汎化能力を示している。新しいトークンデータに基づいてモデルを微調整し、特定のアプリケーションの要件に応じて使用することが可能である。本論文では多音字の認識モデルに焦点を当てており、モデルの微調整は行わず、公開モデルを直接使用して

いる。

・発音のみの手法

モデルの学習曲線は図 5.3 を示す。初期段階では、トレーニングデータにおける損失が減少する一方で、検証データの損失も減少している。しかし、epoch26 以降、検証データの損失が増加し始める。これは、epoch26 以降に過学習が発生していることを示唆している。したがって、本実験では epoch26 の時点でのモデルパラメータを最終的なトレーニング結果として採用した。

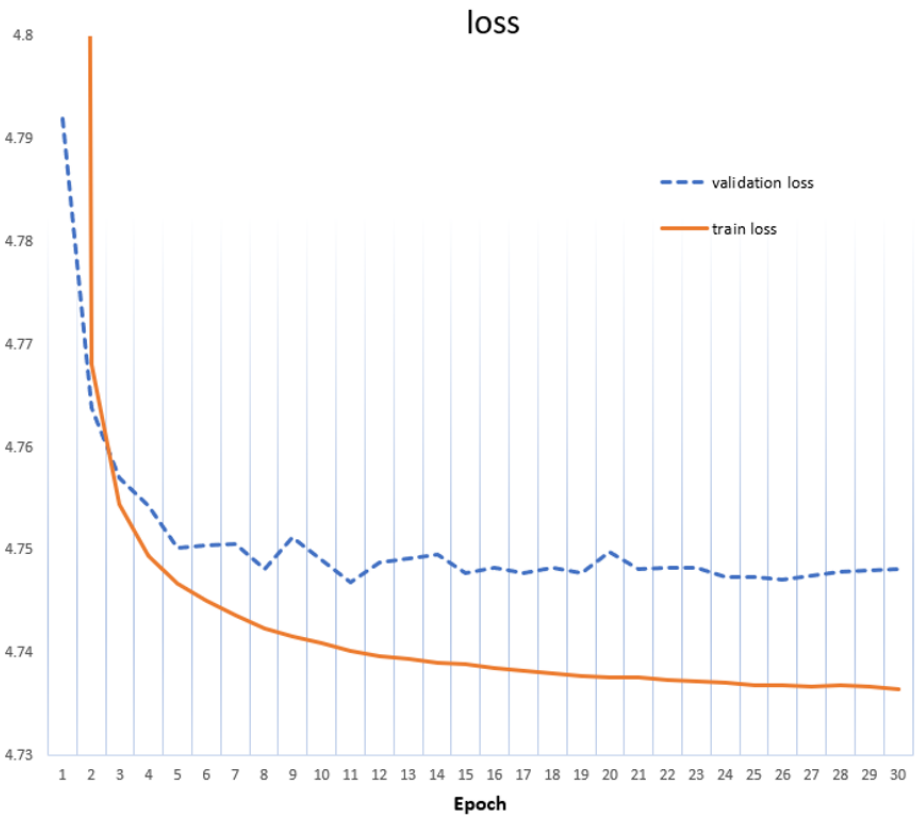

図 5.3: 発音のみの手法の損失関数

・発音と品詞用いる手法

モデルの学習曲線は図 5.4 を示す。初期段階では、トレーニングデータにおける損失が減少する一方で、検証データの損失も減少している。しかし、epoch24 以降、検証データの損失が増加し始める。これは、epoch24 以降に過学習が発生していることを示唆している。したがって、本実験では epoch24 の時点でのモデルパラメータを最終的なトレーニング結果として採用した。

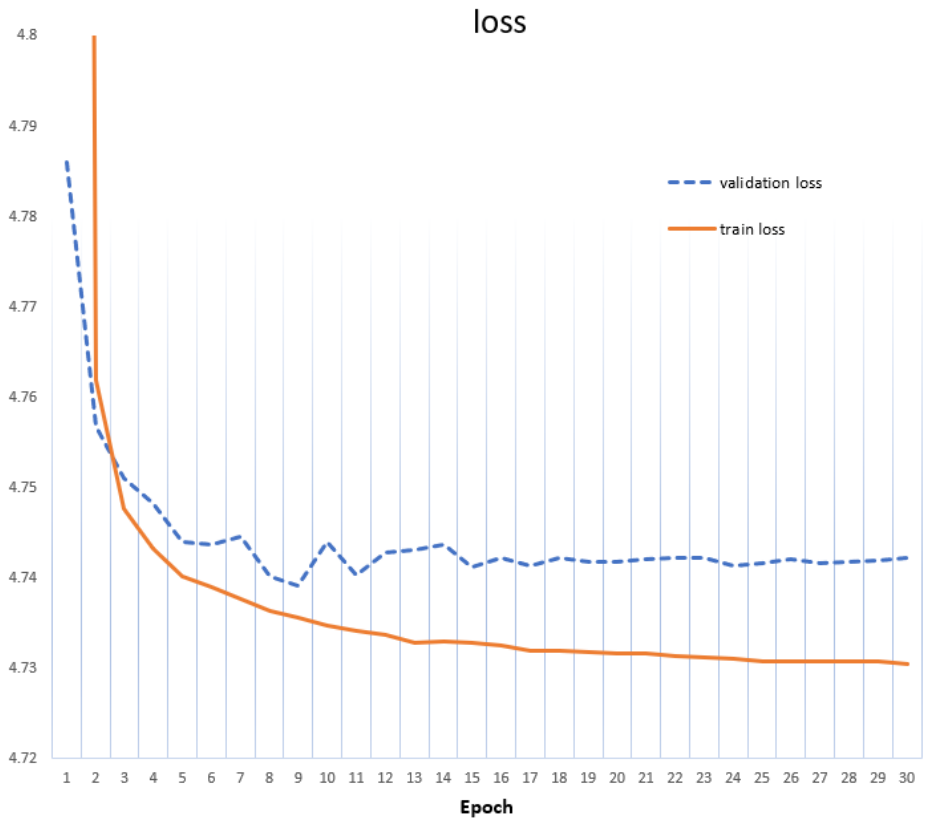

図 5.4: 発音と品詞用いる手法の損失関数

5.3　実験結果

第 4 章の推論プロセスに基づき、トレーニングされたモデルを使用してテストデータに対して推論を行った。読みのみを含む入力方法のモデル精度は 89.23 %であり、読みと品詞を含む方法のモデル精度は 92.37 %であった。モデルの有効性を検証するために、本研究ではランダムベースライン実験を行った。ランダムベースラインとは、特定のモデルやアルゴリズムが関与しない状況下で、単にランダムな選択や生成結果に基づいて比較を行う基準である。ランダムベースライン実験では、ある多音字の読みを予測する際に、その字の固有の読みからランダムに 1 つを選び、それを予測結果とした。結果として、テストデータにおけるランダムベースラインは 44.18 %であった。

この研究結果がランダムベースラインを遥かに上回ることから、提案された多音字の識別モデルの有効性と優位性が示された。また、読みと品詞を含む方法のモデル精度が読みのみの方法よりも高いことから、多音字の品詞情報の入力がモデルの正しい誘導を可能にし、モデルの精度を向上させることに成功したことが示されている。

第 6 章

考察

提案された方法は多音字の効果的な識別が可能であるが、一部の誤った予測が発生することもある。このモデルの性能をより詳細に評価するため、本論文では単一の多音字を対象に分析を行った。

第 5 章で述べた精度は、テストデータ内で正確に読みを予測した回数と総予測回数の比率で計算されるため、単一の多音字に対する評価指標としては適切ではない。そこで、本論文ではテストセット内の各多音字の正確度を計算した。ただし、テストセットには 579 の多音字が含まれる。本論文では、第一の方法において正確度が最も高い 3 つの多音字と最も低い 3 つの多音字を抽出し、さらに正確度が 70 %から 80 %の範囲にある 3 つの多音字をランダムに選び、これらを具体的な研究対象として分析した。各多音字のトレーニングデータが異なる可能性があり、これが識別精度に影響を与える可能性があるため、本論文では各多音字がトレーニングデータ内にいくつ含まれているかも同時に統計した。詳細なデータは表 6 に示されている。

表から明らかなように、正確度が最も高い多音字のトレーニングデータは、最も低い多音字のものよりもはるかに多い。そのため、本研究ではトレーニングデータが正確度に非常に大きな影響を与えると結論付けられる。

本論文では、第一の方法と第二の方法の結果を比較し、その分析を行った。最も正確度が高い 3 つの多音字において、両方の方法の精度は同等であり、これはトレーニングデータが豊富であることから、モデルが基本的に多音字の発音パターンを習得しているため、品詞情報の補助なしでもモデルが多音字を正確に識別できる可能性があることを示唆している。中間の正確度を持つ 3 つの多音字において、両方の方法の結果には大きな差があり、その中で 2 つの多音字は第二の方法での精度が高いことから、品詞を追加した入力

表 6.1: 実験データ

	多音字	Test 個数	方法 1 正確個数	方法 1 正確率	方法 2 正確個数	方法 2 正確率	Train 個数
正確率最高 3 個	有	2208	2207	99.95%	2207	99.95%	6532
	這	1363	1362	99.93%	1362	99.93%	3805
	家	1257	1256	99.92%	1256	99.92%	3675
正確率中位 3 個	服	456	342	75%	401	87%	1290
	参	567	398	70%	452	79%	21235
	燕	185	134	72%	134	72%	1956
正確率最低 3 個	哦	1	0	0	0	0	4
	扛	1	0	0	0	0	5
	爪	1	0	0	0	0	7

方法がモデルの精度を向上させるのに役立つことが示唆されている。これは、これらの多音字が所属する単語の品詞がその発音と強く関連しており、品詞情報がモデルに正しい発音を識別させるのに効果的であることを間接的に反映している。正確度が低い 3 つの多音字において、両方の方法の精度は同等である。これは、この時点でのトレーニングデータが少なく、モデルが多音字の特徴を正確に抽出するのが難しいため、モデルが多音字の発音を正確に識別できない可能性があることを示唆している。この多音字のトレーニングデータを増やすことで、その字の識別精度を向上させる可能性がある。

第 7 章

結論

7.1 まとめ

1. 新しい多音字識別方法

   本研究では、効果的なモデル入力を実現するために、新たな方法を採用した。すなわち、ピンインをテキストに直接組み込み、モデル入力とする手法である。この戦略の主要な目的は、モデル入力内容を簡略化し、不要な情報がモデルの性能に与える影響を軽減することにある。モデルの設計においては、セルフアテンションを基礎とし、多音字の認識に特化した構造を採用し、訓練プロセスを最適化した。また、多音字の精度向上のため、単語属性を有する多音字認識モデルも導入された。これにより、本文中の多音字の表現をより包括的に捉えることが可能となった。これらの革新的な設計と手法の導入により、モデルは多音字の認識において強力な能力を示し、モデル全体の精度と堅牢性が向上するという確固たる基盤が築かれた。

2. モデルの効果の検証

   本研究では、提案された手法の効果を総合的に検証するため、一連の実験を実施した。比較実験の結果からは、単語属性を有する多音字認識手法が、初期の手法よりも多音字の精度において優れていることが示された。これらの実験結果は、提案手法の有用性を裏付けるとともに、特定のタスクに適した多音字認識モデルの設計選択のための明確な指針を提供する。これらの成果は、提案手法の有効性を証明し、モデルの性能向上及び関連する問題解決への強力な支援を提供する。

3. モデルの長所と限界

   実験結果の詳細な分析により、特定の多音文字に関する訓練データの不足が、モ

デルの認識精度における主要な課題であることが判明した。この結果、モデルは特定の文脈において不十分な性能を示す可能性がある。しかしながら、特定の多音字の認識精度が 99% を超える場合も観察された。これは、本研究で設計されたモデルが、特定の条件下で優れた精度を達成していることを示唆している。故に、多音字の正確な識別には、信頼性の高い基盤が提供されていると言える。

品詞を含める方法と含めない方法の実験結果を比較したところ、特定の多音字の識別において品詞の導入がモデルの性能向上に寄与していることが明らかになった。しかしながら、一部の多音字については品詞を使用しなくても正確に識別される場合がある。この観測結果からは、多音字識別における差異化された要求が浮き彫りになり、異なる多音字に対して、一部は品詞の導入により恩恵を受ける可能性がある一方で、他の一部はそれを必要としないことが示唆される。この発見は、モデル改善への新たなアプローチを提供するのみならず、特定の多音字をより正確に識別するための今後の研究に有用な情報を提供する。これらの実験結果の詳細な分析により、モデルの性能差異をより包括的に理解し、さらなる最適化と発展に向けて重要な支援を得ることができた。

7.2　今後の予定

1. モデル構造のハイパーパラメータの検討

   本論文では、新しい入力方法が多音字認識に与える影響を探求することを目的としている。モデルトレーニングの複雑性を低減するため、BERT のコア構造に基づく簡略化されたモデルを採用した。このモデルは 6 つの attention layer を含む。しかし、モデル構造は性能に大きな影響を及ぼす重要な要素であるため、モデル構造の選択と最適化は避けられない課題である。したがって、モデル構造のハイパーパラメータ（例えば、層数やノード数）のさらなる研究が、将来のモデル改善に向けた新たな戦略を提供する可能性があると考えている。今後の研究では、モデル構造が多音字認識の性能に与える影響をより深く探究し、ハイパーパラメータの調整を通じてモデルを最適化することを目指す。このアプローチは将来の研究の重要な方向性を示し、モデルの精度と適用性のさらなる向上を目指す。
2. 特定多音字のデータ量とモデルの精度の関係についての定量分析

本論文において、異なる多音字に関する実験結果を分析し、トレーニングデータ量が多音字の認識精度に一定の影響を及ぼすことを初歩的に示した。しかし、特定の多音字における異なるデータ量がモデルの精度に具体的にどのように影響するかについての詳細な対照実験は行われていない。より詳細な実験設計を通じて、より正確な分析を実施し、モデルのトレーニングデータ量と多音字の認識精度の関連性を深く理解することが可能である。これにより、特定の多音字の認識精度を向上させるためのより効果的な戦略を提供することができる。したがって、このような実験分析は将来の研究課題として重要であり、特定多音字の認識精度向上のためのより精確な戦略を提供することが期待される。

謝辞

最後に，本論文を作成するにあたり指導・助言を頂いた，指導教官の新納浩幸教授に心より感謝申し上げます．また，普段の勉強会にて指摘・助言を頂いた新納研究室の皆さんに感謝致します．

参考文献

[1] 孔祥卿, 史建偉, 孫易, 白艷 and others. 漢字學通論. BEIJING BOOK CO. INC., 2020.

[2] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, Vol. 323, No. 6088, pp. 533–536, 1986.

[3] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, Vol. 9, No. 8, pp. 1735–1780, 1997.

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, Vol. 30, , 2017.

[5] Daju Gou and Wanbo Luo. Processing of polyphone character in chinese tts system. *Chinese Information*, Vol. 1, pp. 33–36.

[6] Hong Zhang, JiangSheng Yu, WeiDong Zhan, and Shiwen Yu. Disambiguation of chinese polyphonic characters. In *The First International Workshop on MultiMedia Annotation (MMA2001)*, Vol. 1, pp. 30–1. Citeseer, 2001.

[7] Changhao Shan, Lei Xie, and Kaisheng Yao. A bi-directional lstm approach for polyphone disambiguation in mandarin chinese. In *2016 10th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, pp. 1–5. IEEE, 2016.

[8] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[9] Zhiheng Huang, Wei Xu, and Kai Yu. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015.

[10] Kyubyong Park and Seanie Lee. g2pm: A neural grapheme-to-phoneme conversion package for mandarin chinese based on a new open benchmark dataset. *arXiv preprint arXiv:2004.03136*, 2020.

[11] Haiteng Zhang. Polyphone disambiguation in chinese by using flat. In *Interspeech*, pp. 4099–4103, 2021.

[12] 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.

[13] Yi-Chang Chen, Yu-Chuan Chang, Yen-Cheng Chang, and Yi-Ren Yeh. g2pw: A conditional weighted softmax bert for polyphone disambiguation in mandarin. *arXiv preprint arXiv:2203.10430*, 2022.