令和 5 年 2 月 20 日 (水) 修士論文発表会

深層学習 による灯謎問題の正解推定システムの構築

創発ソフトウェア研究室

M2　陳　偉斉

発表の構成

1.はじめに

2.要素技術

3.データセット

4.提案手法

5.実験

6.まとめと今後の課題

発表の構成

1.はじめに

2.要素技術

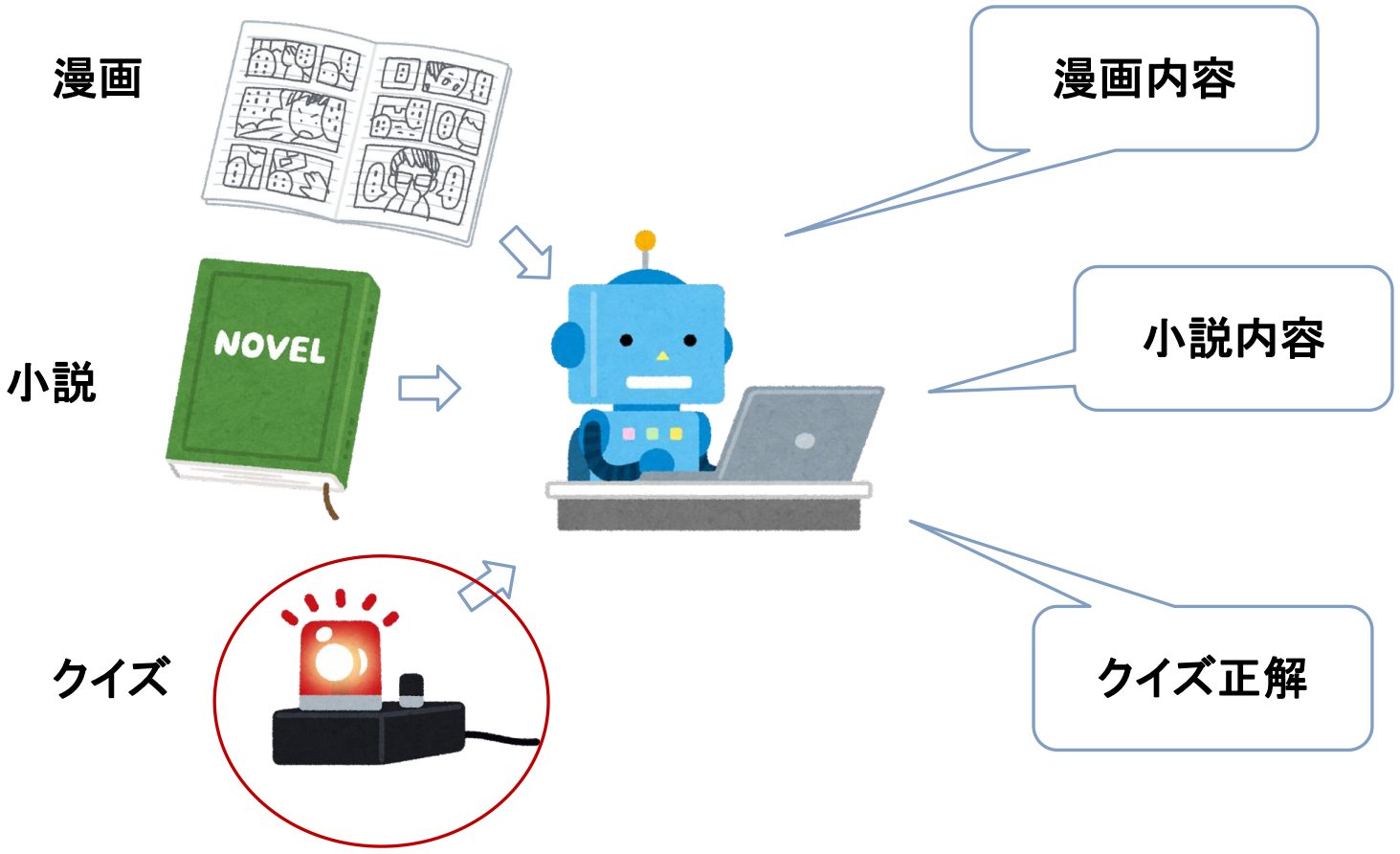
3.データセット

4.提案手法

5.実験

6.まとめと今後の課題

はじめに

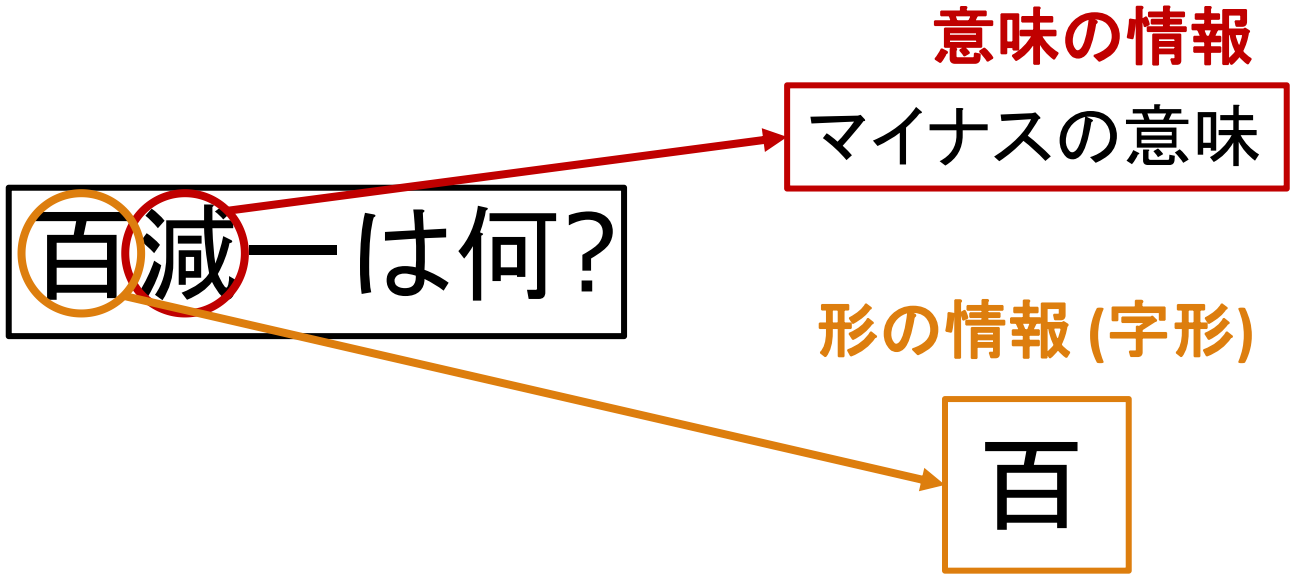


灯謎

謎面（問題）	謎目（ヒント）	謎底（答え）
百減一は何?	答えは一文字	白

- 「問題」,「ヒント」,「答え」のセット
- 文字の形に注目する種類の問題のみ使用

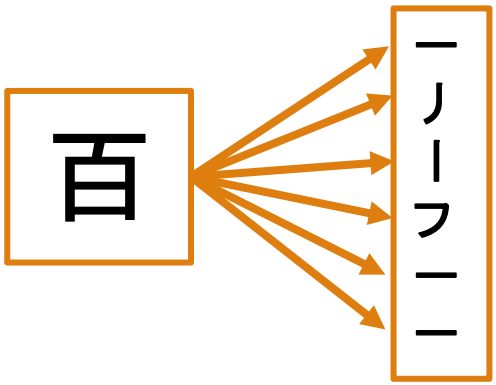
漢字の意味の情報と形の情報



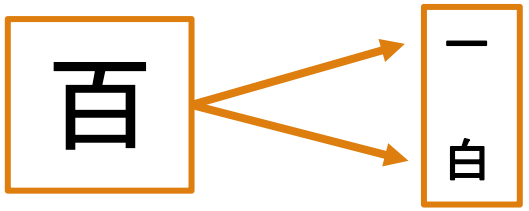
・灯謎を解くには，漢字の**意味の情報**と**形の情報**が必要

漢字の形の情報

漢字の画



SUB 漢字



・本研究は漢字の画のみを対象

はじめに

研究目標

人工知能による灯謎の解答生成システムの構築

本研究の課題

灯謎の正解推定システムの構築

灯謎のデータセットの構築

発表の構成

1.はじめに

2.要素技術

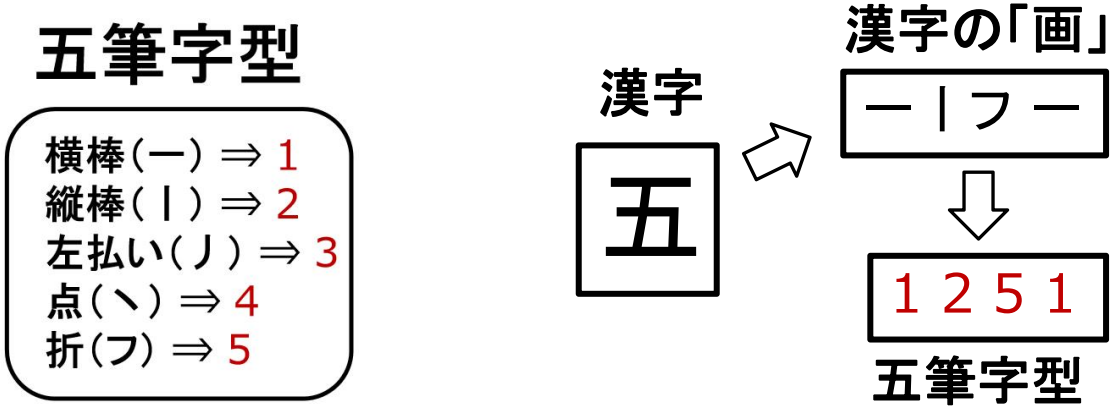
3.データセット

4.提案手法

5.実験

6.まとめと今後の課題

五筆字型入力方法



漢字の画を「横棒」,「縦棒」,「左払い」,「点」,「折」に分類し, それぞれ数字 1 から 5 に割当

生成した数字の列を「五筆字型」と呼称

- 五筆字型計算机漢字輸入技術. 河南科学技術出版社, 1985.

Levenshtein 距離

- Levenshtein 距離は 2 つの文字列の
最小編集回数を表示
- 本研究は漢字間の違いを表現するために使用

文字列「JULY」から「JELLY」に変形する手順

「JULY」

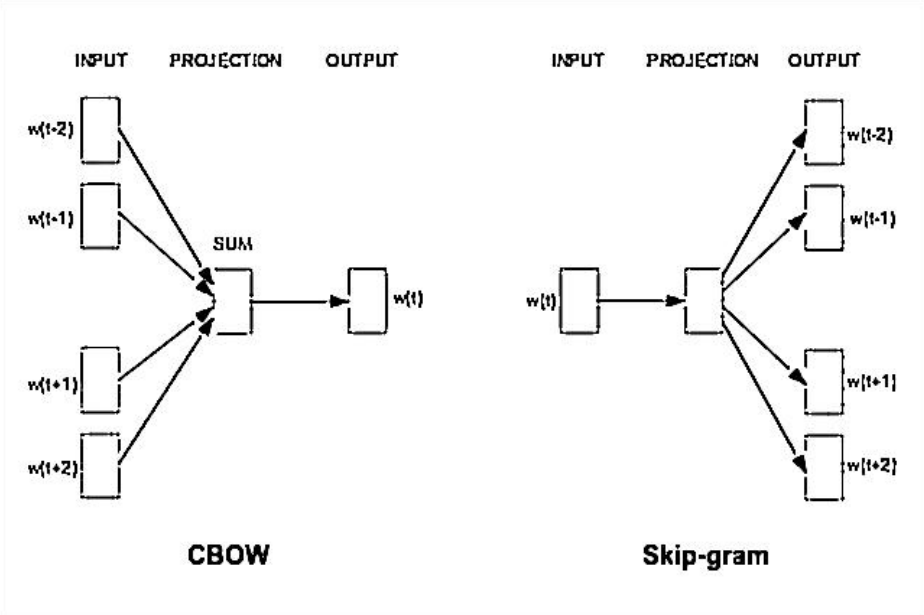
Step1.「JELY」（「U」を「E」に置換）

Step2.「JELLY」（「L」を挿入して終了）

Levenshtein 距離 = 2

・V. I. Levenshtein, et al. Binary codes capable of correcting deletions, insertions, and reversals. In Soviet physics doklady, Vol. 10, pp. 707–710. Soviet Union, 1966.

Word2Vec

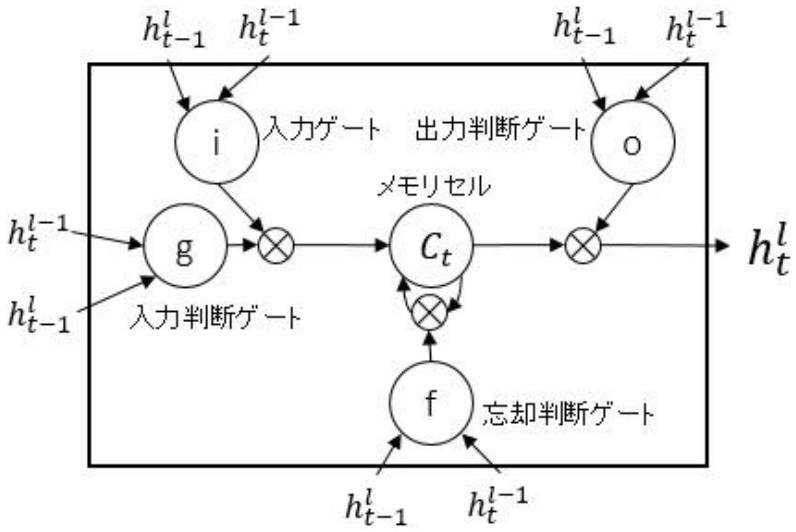


・漢字ベクトルの生成に使用

・事前学習済みのモデル“Chinese Word Vectors”を使用

・T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. In Y. Bengio and Y. LeCun eds., 1st Interna_x0002_tional Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings, 2013.

Long Short-term Memory (LSTM)



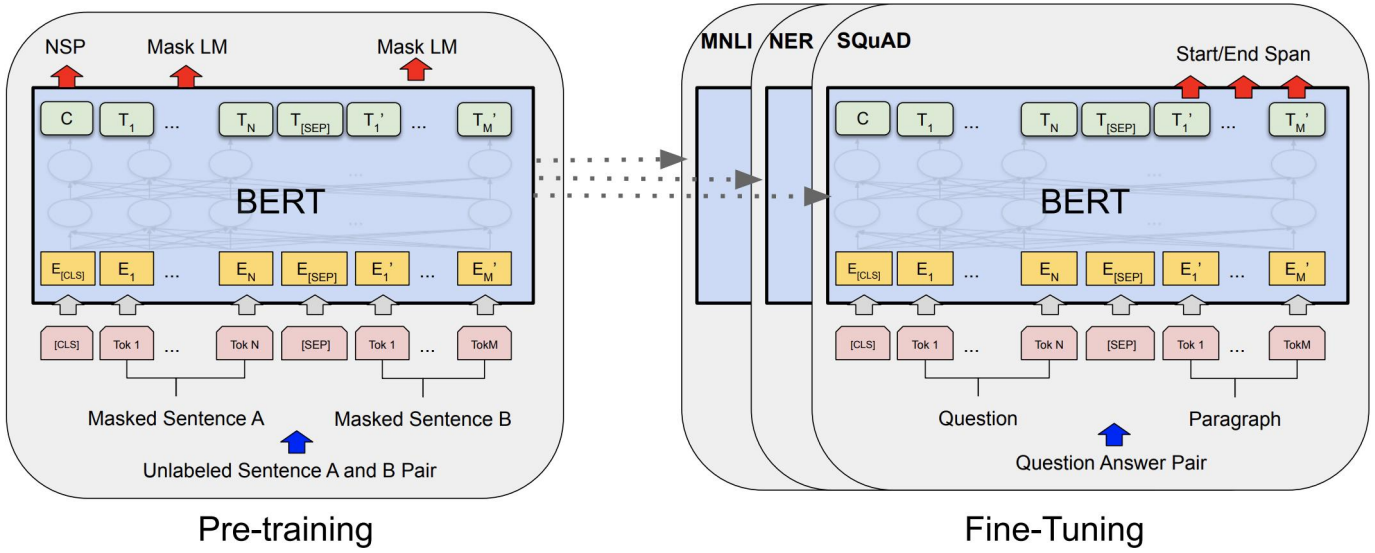
・ゲート構造で勾配を制御

・メモリセルcで情報を記憶

・長期な記憶が可能

・W. Zaremba, I. Sutskever, and O. Vinyals. Recurrent neural network regularization. CoRR, abs/1409.2329, 2014.

Bidirectional Encoder Representation from Transformers (BERT)



・事前学習済みの BERT モデル “bert-base-chinese” を使用

・J. Devlin, M.-W. Chang, K. Lee, and K. 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, June 2019.

発表の構成

1.はじめに

2.要素技術

3.データセット

4.提案手法

5.実験

6.まとめと今後の課題

データセット構築の手順

- 中華灯謎データベースでは中国各地の灯謎愛好者が収集，作成した灯謎問題 1,408,684 件を収録
- 本研究は収録した灯謎の問題，ヒント，答えのみを使用
  「**問題 + ヒント + 答え**」形式のデータを**正解データ**と定義
- 研究対象は**答えが一文字**であり，**文字の形に注目**する灯謎問題 (72,937件)

・[中华灯谜库]: http://www.zhgc.com/mk/index.asp.

データセット構築の手順

- 以降では，便宜的に以下のように呼称

正解データ：灯謎データベースから収集したデータ

問題 ＋ ヒント ＋ 本来の正解を表示する漢字

正解データセット：正解データで構成するデータセット

不正解データ：同じ問題とヒントに対して別の答えを付与したデータ

問題 ＋ ヒント ＋ 人為的設定した不正解を表示する漢字

不正解データセット：不正解データで構成するデータセット

不正解データセット構築の手順

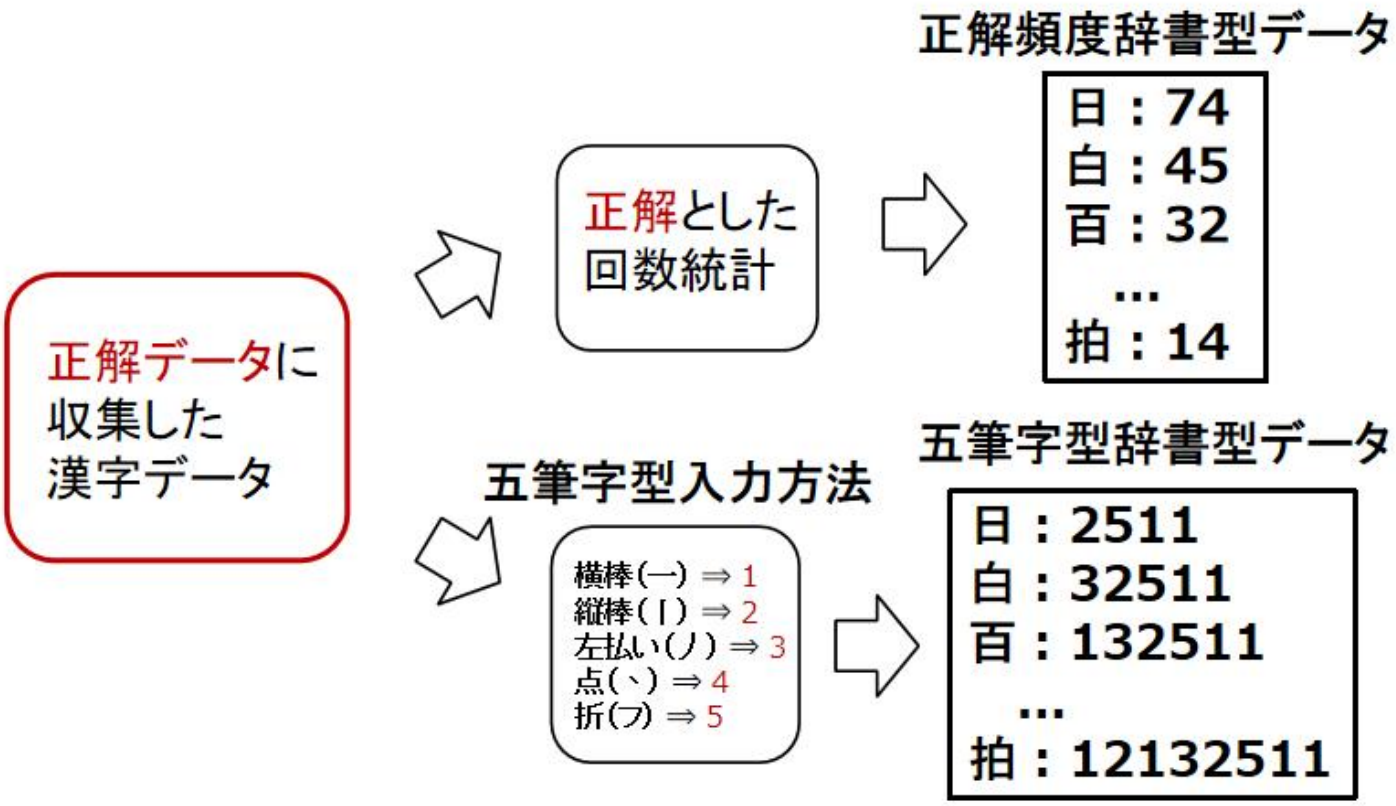
- 正解データによる漢字を辞書型データの作成

1. 正解データから漢字を収集

2. 答えとして使用される漢字の頻度の
   辞書型データを作成

3. 五筆入力方法により五筆字型の
   辞書型データを作成

辞書型データ生成の例


正解データに
収集した
漢字データ
正解とした
回数統計
正解頻度辞書型データ
日：74
白：45
百：32
…
拍：14
五筆字型入力方法
横棒(一) ⇒ 1
縦棒(丨) ⇒ 2
左払い(丿) ⇒ 3
点(丶) ⇒ 4
折(乛) ⇒ 5
五筆字型辞書型データ
日：2511
白：32511
百：132511
…
拍：12132511

不正解データセット構築の手順

- 以降では，便宜的に以下のように呼称

難易度の低い漢字：Levenshtein 距離が大きい漢字

不正解データセット (難易度低い)：
「問題 ＋ ヒント ＋ 難易度の低い漢字」形式のデータセットで構成する不正解データセット

難易度の高い漢字：Levenshtein 距離が小さい漢字

不正解データセット (難易度高い) ：
「問題 ＋ ヒント ＋ 難易度の高い漢字」形式のデータセットで構成する不正解データセット

不正解データセット構築の手順

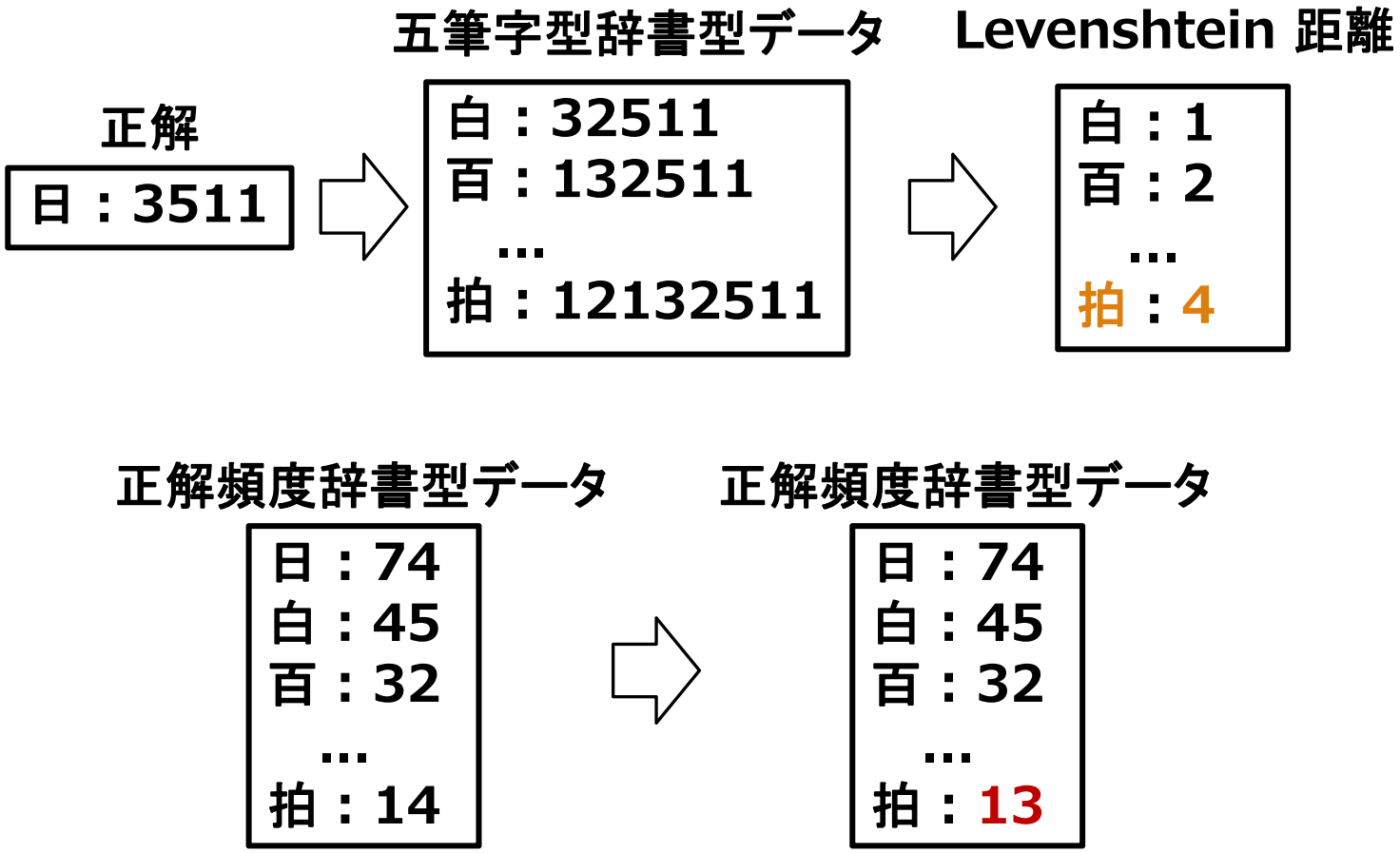
- 不正解データを表示する漢字の難易度設定

1. 正解漢字と全部漢字の五筆字型間の Levenshtein 距離を計算

1. 正解漢字に対して, 難易度の低い漢字と難易度の高い漢字を設定

1. 不正解データセット (難易度低い) と不正解データセット (難易度高い) を構築

不正解漢字設定（難易度低いの例）



データセットの統計

データセット	データ数
正解データ	72,937
不正解データ (難易度低い)	72,937
不正解データ (難易度高い)	72,937

- 正解と不正解のデータ数を揃えた

発表の構成

1.はじめに

2.要素技術

3.データセット

4.提案手法

5.実験

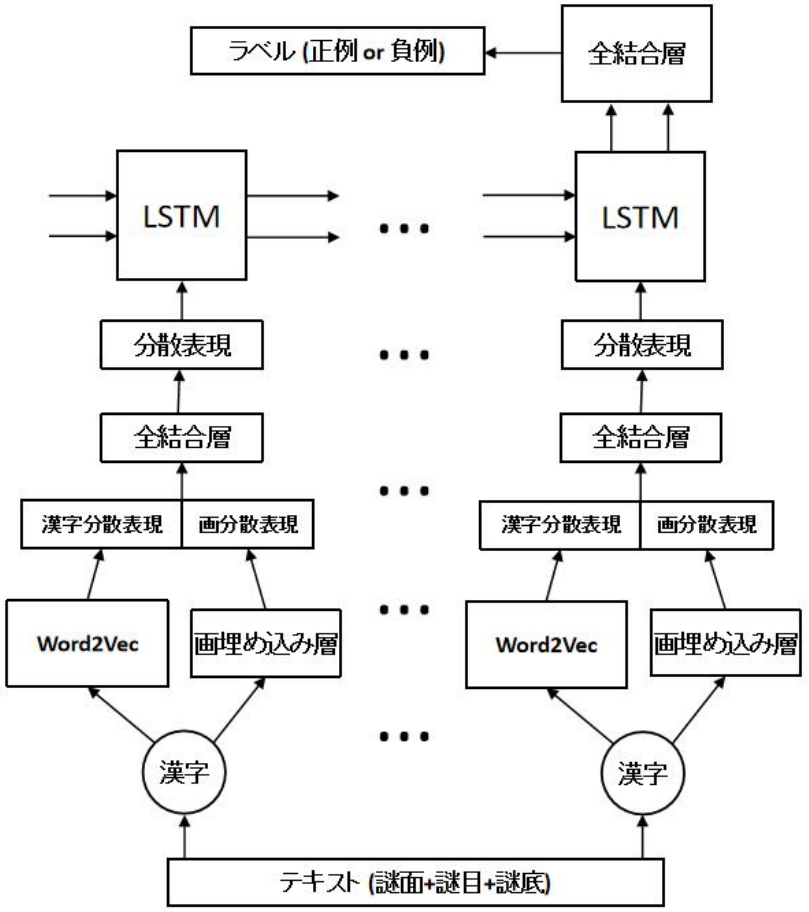
6.まとめと今後の課題

提案手法

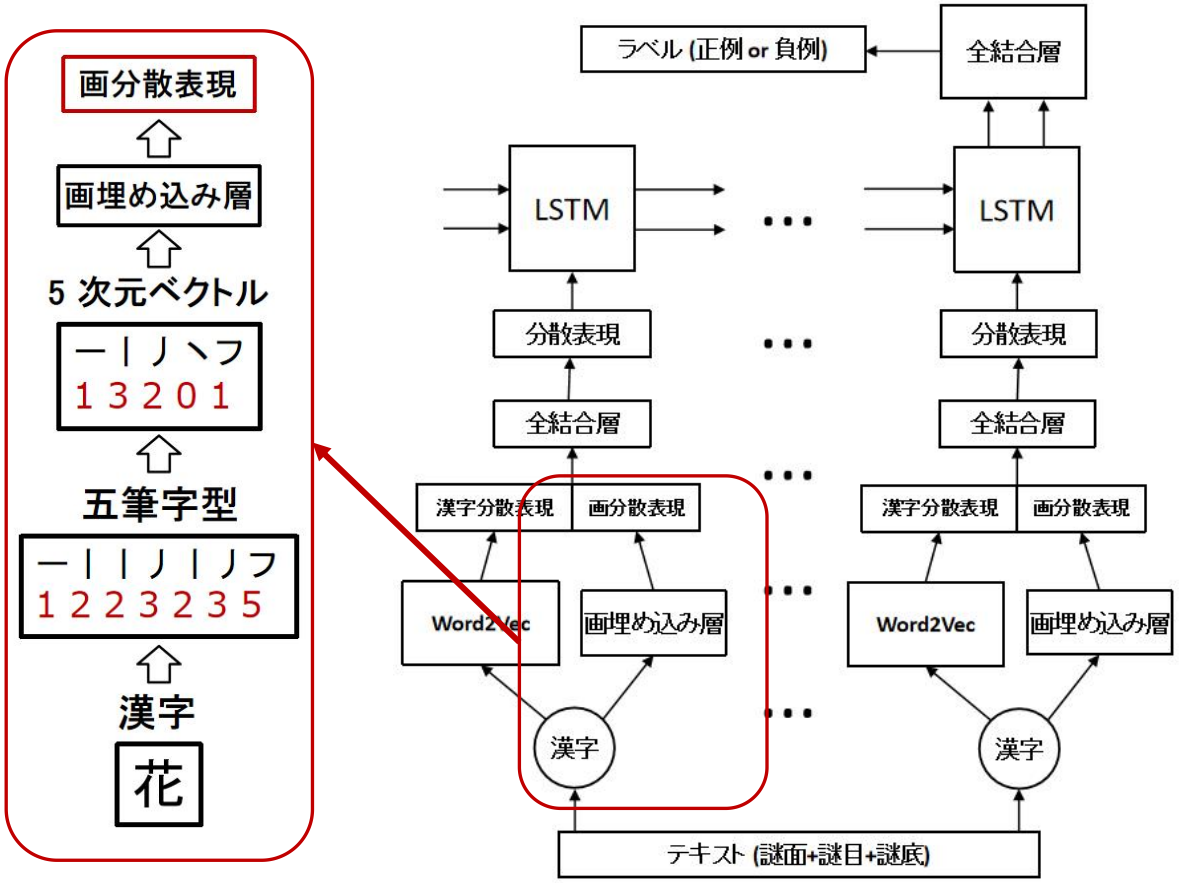
- 漢字の画情報を利用した灯謎問題正解推定モデル

対象とする灯謎問題の
「問題 + ヒント + 答え」部分を
順次に入力, データの前文により
答えは正解か不正解かを推定

漢字の分散表現生成に
漢字の画情報を導入

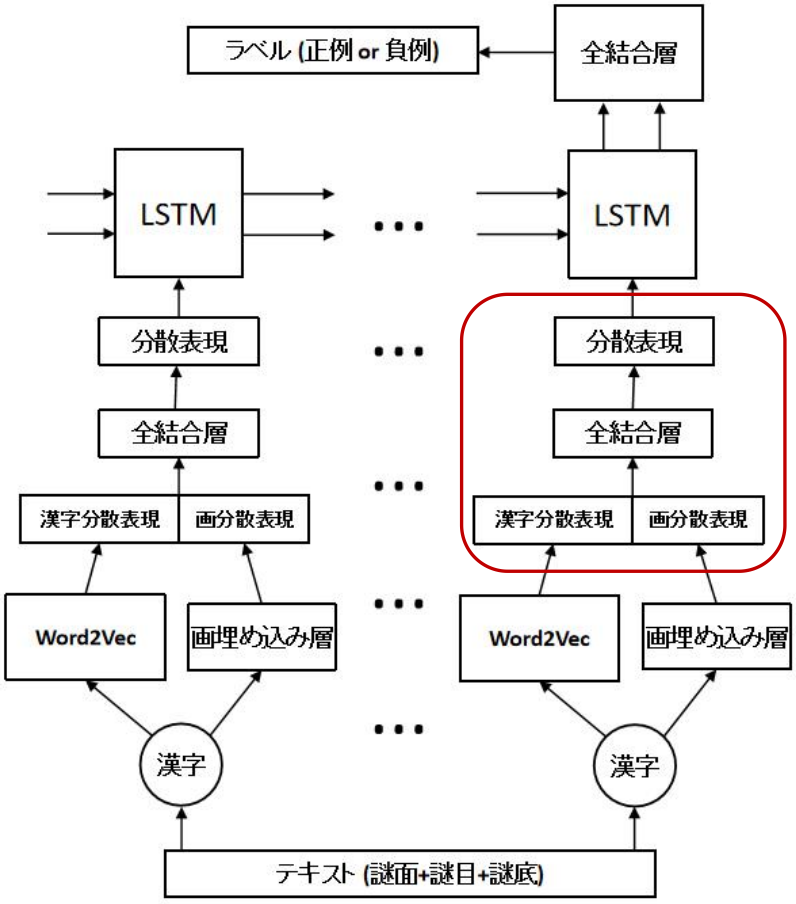


漢字の画の分散表現の生成



- 5 次元ベクトルは漢字を構成の「横棒」,「縦棒」,「左払い」,「点」,「折」の出現回数を表示

画情報を含まれた分散表現の生成



- 漢字分散表現と画分散表現を結合
- 漢字分散表現と分散表現は同じ 300 次元

発表の構成

1.はじめに

2.要素技術

3.データセット

4.提案手法

5.実験

6.まとめと今後の課題

実験説明

- 実験用データセットは 3 つの手法を使用

  BERT+MLP：

  事前学習済みの BERT モデル “bert-base-chinese” を使用

  BERTモデルで灯謎データの分散表現を生成し，2 値分類 MLP で

  灯謎の答えは正解か不正解かを推定

  LSTM+Word2Vec：

  事前学習済みの “Chinse Word Vectors” を使用

  LSTM で前文との関係を考慮した分散表現を生成し，

  最後に全結合層に入力して灯謎の答えは正解か不正解かを推定

  LSTM+Word2Vec+Stroke：

  本研究の提案手法, LSTM+Word2Vec 手法の上漢字の画情報を考慮

実験説明

- 実験用データセットは 2 つを使用

  以降では以下のように呼称
  難易度の低いデータセット:
  正解データセット + 不正解データセット (難易度低い)

  難易度の高いデータセット:
  正解データセット + 不正解データセット (難易度高い)

- 入力データは「**問題;ヒント;答え**」形式 (**;** は分割符号)

- 比較するために, 画情報入れ込まない LSTM + Word2Vec と BERT + MLP モデルを使用

実験データの処理

難易度の低いデータセット	訓練データ	テストデータ
データ総数	106,464	26,616
正解データ	53,277	13,263
不正解データ	53,187	13,353
難易度の高いデータセット	**訓練データ**	**テストデータ**
データ総数	106,464	26,616
正解データ	53,277	13,263
不正解データ	53,187	13,353

・実験用データを 8 対 2 で訓練データ, テストデータに分割

・2 つのデータセットは正解データは同じであり, 不正解データのみ不同

提案手法の実験条件

パラメータ	数値
分散表現の次元数	300
画分散表現の次元数	30
隠れ層の次元数	256
バッチサイズ	128
Dropout	0.5
損失関数	Cross-Entropy　Loss
最適化手法	Adam
学習率	0.00003
Epoch 数	400

・画情報入れ込まない LSTM + Word2Vec は同じパラメータを使用

BERTの実験条件

パラメータ	数値
隠れ層の次元数	256
バッチサイズ	128
Dropout	0.5
損失関数	Cross-Entropy　Loss
最適化手法	AdamW
学習率	0.00005
Epoch 数	400

実験結果

難易度の低いデータセットによる実験結果

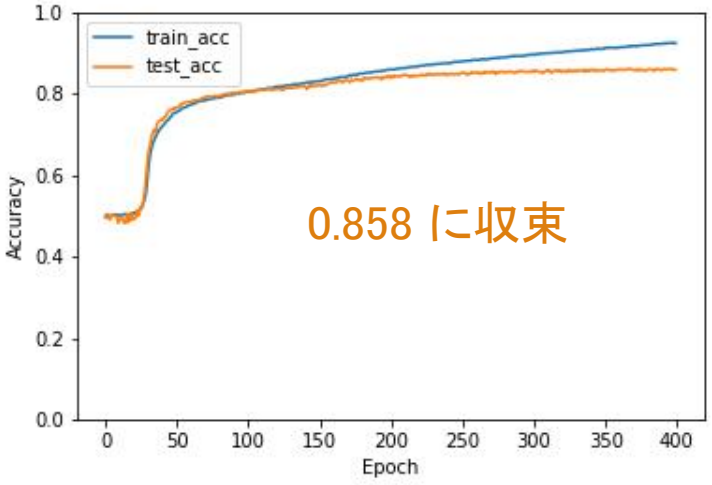
手法	Accuracy	Pricision	Recall	F1 値
BERT+MLP	73.26%	74.08%	72.40%	73.22%
LSTM+Word2Vec	81.68%	79.73%	84.78%	82.18%
LSTM+Word2Vec+Stroke	**85.76%**	**85.55%**	**85.94%**	**85.74%**

難易度の高いデータセットによる実験結果

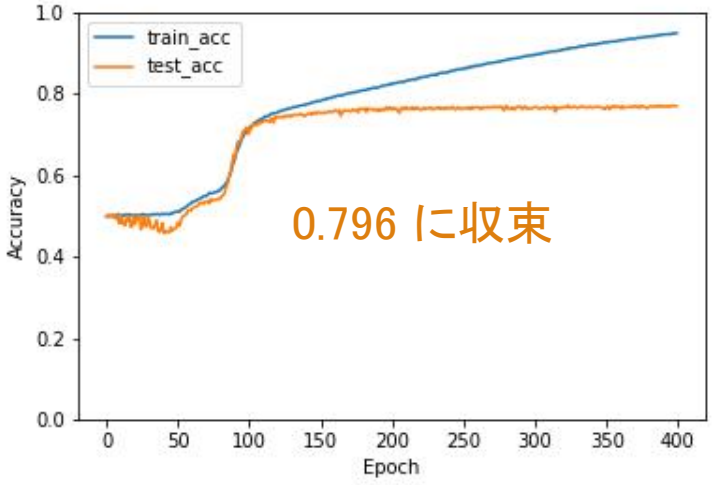
手法	Accuracy	Pricision	Recall	F1 値
BERT+MLP	61.53%	61.13%	61.53%	61.33%
LSTM+Word2Vec	76.88%	75.80%	78.70%	77.22%
LSTM+Word2Vec+Stroke	**79.61%**	**78.76%**	**80.89%**	**79.81%**

- 結果として提案手法は比較的に有効

提案手法によるテストデータの精度推移



難易度の低いデータセットの精度推移



難易度の高いデータセットの精度推移

- 人工知能の灯謎の対する能力は人類と類似

テストデータの例

問題: 十八学士 (唐代十八人の政治家の意味)
ヒント: 7 筆字 (答えは 7 画の漢字)
正解: 李
(「十八」は「木」を表し,「学士」は「子」を表現
故に「 十 + 八 + 子 = 李」が正解)
難易度の高い漢字: 季 (正解に高類似性)
難易度の低い漢字: 惨 (正解に低類似性)

- 提案手法は推定成功が, 他の手法では推定失敗

発表の構成

1.はじめに

2.要素技術

3.データセット

4.提案手法

5.実験

6.まとめと今後の課題

まとめと今後の課題

- まとめ

  灯謎の正解推定システムの構築

  灯謎データセットの構築

  画情報の導入は灯謎の正解推定に有効

- 今後の課題

  モデルの精度向上の考案

  別の種類の灯謎を用いてデータセットを構築

  人工知能による灯謎の解答生成システムの構築

ご清聴ありがとうございました