人工知能による灯謎問題解決のため部首情報を含めた漢字分散表現生成手法の調査

1　はじめに

質問応答タスクは文書をもとに入力された質問に対して正しく応答することを目的とするタスクである. 今まで質問応答に対する研究は読解問題と呼ばれ, 主に問題と答えの情報を含む長文を対象としてきた. その一方で, 問題の中に隠された情報で問題を解けるクイズ問題と呼ばれタスクも存在する. 本研究ではこのクイズ問題にとりくむ. 具体的には人工知能で中国の伝統的クイズ問題「灯謎 (トウメイ)」を解く方法について考える.

2　灯謎 (トウメイ)

灯謎は, 中国の伝統的クイズ問題である. 質問者は問題を詩や熟語の形で出し, 回答者はそれぞれ回答する. 答えは常に字または単語になる. 質問応答とは違い, 灯謎は質問に答えるための問題文以外の文書や知識など必要がなく, 質問の文中から答えの情報を得る. 言い換えと, 質問を理解すれば回答できる. 灯謎を解くためには, 問題に隠された情報をもとに, 問われている内容を理解して抽出しなければならないので, 灯謎の研究は一種の情報抽出として考えることもできる.

灯謎のパターンはだいたい, 謎とヒントと答えで構成される. 謎は詩や熟語, あるいは普通の話し言葉で記述された文である. ヒントは答えの形を説明する文である. ヒントは 1 つ以上与えられる答えは字か単語であり問題に隠された字の構成, 発音, 意味などの情報からで解くことができる. 図 1 に灯謎の一つの例を示す.

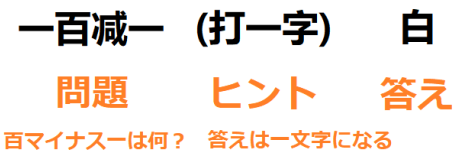

図 1: 灯謎の例

灯謎問題のうち, 字謎と呼ばれる答えが一つの漢字のみとなる種類のみについて考える. 字謎の答えは, 単語の意味に加えて漢字の形も強く関わるので, 単純に大量の問題の文の情報のみをニューラルネットワークでがくしゅうしても意味がない. そこで本研究では漢字の形に着目し, 漢字の部首情報を考慮した,CharCBOW と CharSkipGram モデルで問題文から字の情報を得られるようにする.

3　要素技術

3.1　分散表現

分散表現, あるいは単語埋め込みとは, 単語を比較的小さい次元の実数ベクトルで表現する技術である. 機械には人間の言葉のような自然言語における意味情報を理解することが難しいため, 自然言語を分散表現の形変換することで意味情報を扱いやすくする. 現在, よく使われている分散表現の手法は Word2vec, ELMo, BERT などがある. 本研究では Word2vec の手法を利用する. 下の図 2 に単語の分散表現の例を示す.

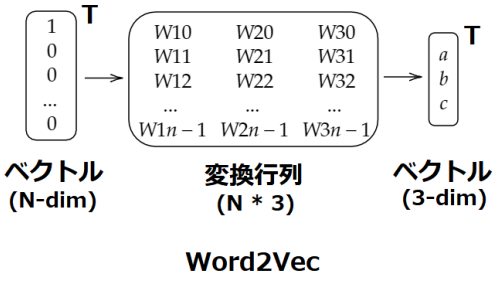

図 2: 単語の分散表現の例

3.2　One hot encoding

One hot encoding は, 字や単語などの入力データを語彙の長さと同じ次元数があるベクトルに変換する手法である. ベクトル中の入力データに対応する次元が 1, それ以外は全部 0 とする.

しかし, この手法は二つの問題がある. 一つは One hot encoding の計算量が語彙の長さに比例して大きく

なること, もう一つはベクトルの間に意味的関連性がないことである.

3.3 Word2vec

Word2vec[1] は,Mikolov らが 2013 に提案したモデルである.

Word2vec は, 単語の分散表現を生成するためのアルゴリズムである. 二層のニューラルネットワークのみで構成されるという特徴により, モデルの計算量は比較的少なくなり, 大規模なデータで分散表現を学習することが可能となる.Word2vec には CBoW(continuous bag-of-word) モデルおよび skip-gram モデルの二つのモデル構造があり, そのいずれかを用いて, データの分散表現を生成する.

CBoW では, 入力として周囲の単語を与え, その中心の単語の予測を出力する. この学習を通じて, ネットワークは, 周囲の単語から中心の単語としてどのような単語が現れる可能性が高いかを学習する.

CBoW と違い,skip-gram は目標とある中心の単語から周囲の単語を予測する手法である. skip-gram の場合, 入力は中心の単語, 出力は周囲の単語となる.

モデル構造が違うため,CBoW はデータ量が十分である場合に適用するとよく, よく用いられる単語に対して学習した場合良い結果が得られる. それに対して,skip-gram はデータ量が不十分である場合に適用するとよく, あまり頻出しない単語に効果的である. 図 3 に CBoW および skip gram モデルの構造を示す.

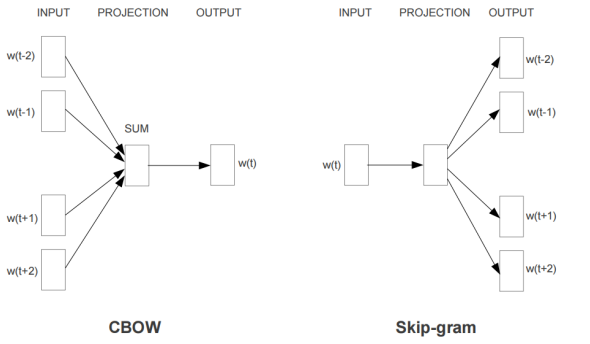

図 3: Word2vec[1] の構造

3.4 charCBOW と charSkipGram

分散表現に関する研究において, 分散表現の単位は語や句や文などさまざまである. 中国語に関する研究では主に単語や漢字を文脈の最小単位として扱っている. しかし, 漢字の中に含まれる情報に注目する研究は少ない. 漢字は, 部首と偏で構成されている, この中には漢字の意味を提示する情報も含まれている. 例えば, 漢字「花」と「草」の部首には植物の意味が含まれている. この部分の情報は質問応答や感情分析などのタスクに利用できる可能性がある. CharCBOW と CharSkipGram[2] は Yanran らが 2015 に提案した, 漢字と漢字の部分の漢字埋め込み方法である. この手法は Word2vec の手法に基づいており, CharCBOW の入力では周囲の単語の漢字と漢字の部分情報の結合ベクトルである. そして結合ベクトルを入力とし, 中心の単語を予測する.CharSkipGram は CharCBOW と違い, 中心の単語から周囲の単語の漢字結合ベクトルを予測する形である.

図 4 に CharCBOW および CharSkipGram の構造を示す.

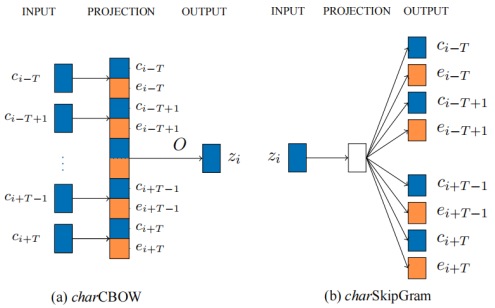

図 4: CharCBOW と CharSkipGram[2]

4 データセット

本研究では Chinese Wikipedia Dump データセット [3] (zhwiki) で抽出した 10 記事で CBoW および CharCBOW モデルを学習する. 学習する前データセットの文は全部漢字に分ける. そしてオンライン新華辞書 [4] を利用し, CharCBOW に用いる部首データセットを構築する.

灯謎の実験に対して, 中華灯謎庫 [5] より答えが一文字である灯謎 (字謎) を収集し, 灯謎のデータセットを構築した.

5 実験

漢字部首の情報を灯謎問題に使う有効性を探るために実験する.

まず CBoW と CharCBOW モデルで漢字の分散表現を訓練する. そして CBoW と CharCBOW の性能を確認するため, 灯謎データセットの中から答えと答えに対応するヒント漢字を抽出して, その分散表現を出力した. これらの漢字に対する漢字類似度でどちらのモデルが優れているかを確認する.

今回の実験データは zhwiki データセット 1186 文を 50318 字に分け, 漢字の種類の数は 1638 であり, 部首の数は 190 である. 次に, 漢字と部首の数ごとに辞書を構築する. 部首の中で, 意味が近いものは全て同じ部首とした. 表 1 に zhwiki データセットの情報を示す.

表 1: DataSet

データの情報	数値
文の総数	1186
漢字の総数	50318
漢字の種類	1638
部首の	190

最後に漢字と部首のデータを前後文それぞれ 2 個ずつ 4 個を 1 組として 328526 ペアに分けた.

訓練データが多いので, ネガティブサンプリングを使用した CBoW モデルで実験した. そして CharCBOW モデルの実装は CBoW に基づき, 入力データは漢字データと部首データの One hot ベクトルの連結とした. 同時に,CharCBOW のパラメータ行列も拡張する.

図 5 に実験に用いた CharCBOW の構造を示す.

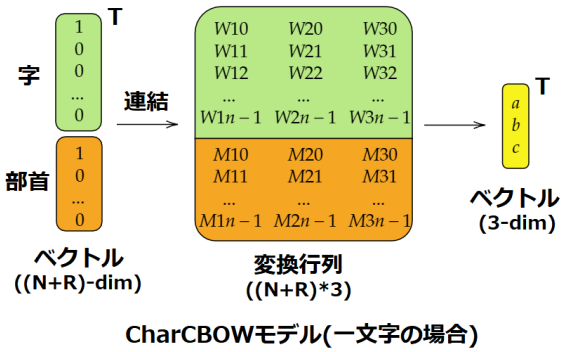

図 5: CharCBOW の例

表 2 に実験パラメータを示す.

漢字ペアをバッチサイズ 10 で 5133 バッチに分けて学習した.

CBoW の訓練誤差は 68.37 に収束し,CharCBOW の訓練誤差は 58.78 に収束した. 図 6 に CBoW 及び CharCBOW モデルの誤差変化曲線を示す.

表 2: モデルのパラメータ

パラメータ	数値
CBOW の入力サイズ	1638
CharCBOW の入力サイズ	1828
分散表現の次元数	100
周囲の単語のサイズ	4
バッチサイズ	64
学習率	0.02
最適化手法	SGD
損失関数	Cross-Entropy

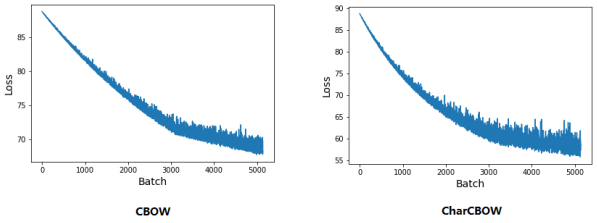

図 6: 誤差曲線

学習に部首の情報が含まれているかどうかを確認するため, 灯謎データセットの中に 10 問題を抽出し, 答えの漢字類似度を計算する. そして計算結果の中に, 上位 10 個の漢字を抽出した. 図 7 に答えが漢字「白」になる場合の漢字類似度を示す.

CBOW	CharCBOW
('白', 1.0000000000000002)	('白', 0.9999999999999999)
('少', 0.3297009962465719)	('蒂', 0.3319461148955747)
('相', 0.3071867132568882)	('拷', 0.3019227779989213)
('制', 0.28220268094104894	('启', 0.27740046353636777
('溯', 0.28098215579229135	('脉', 0.26681677259516035
('亚', 0.28096580839296526	('备', 0.2577849546891706)
('钢', 0.2782392237836553)	('噪', 0.2540897524721133)
('籍', 0.2673191581910037)	('渐', 0.25297692245759795
('迈', 0.2671075324326715)	('录', 0.25186299022406444
('行', 0.2641859663810115)	('给', 0.24796874255955345

図 7: 答えが「白」の場合の漢字類似度

6　まとめと今後の課題

本研究では, 灯謎問題を解決する人工知能の構築のために,CBoW モデルを利用した. そして部首情報を含める CharCBOW モデルを作り, 実験により評価した. 結果として CharCBOW が CBoW 有効だと確認したが, 両方の結果も期待される結果ではないことも確認した. 原因を考えると, まずはデータ量の不足である, そして, 漢字と部首以外の情報が含めていないことも原因として考えられる.

今後の課題は, 漢字の部分と単語の意味を含めている手法,ELMo や BERT を利用した灯謎問題に対する漢字の埋め込み方法を考える. 更に漢字の画像と音声情報も埋め込みに使う可能性を考える.

参考文献

[1] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013.

[2] Yanran Li, Wenjie Li, Fei Sun, and Sujian Li. Component-enhanced Chinese character embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 829–834, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[3] Chinese wikipedia dump. `http://download.wikipedia.com/zhwiki`.

[4] Online xinhua dictionary. `http://xh.5156edu.com/`.

[5] Chinese puzzle database. `http://www.zhgc.com/mk/index.asp`.