ユーザー支援を目的とした音声質問応答システム *

☆松好祐紀, 滝口哲也, 有木康雄 (神戸大)

1　はじめに

情報社会である現代では、職種を問わず機械やプログラムなどのシステムを使う機会が多い。最初の段階ではマニュアルと顔を突き合わせることになるが、それだけで十分使えるようになるのは困難である。機械や情報技術に疎い人ならなおさらである。そこで実際に動かしながら、使いながら慣れていくことが多いと思われる。この際に熟練者のアドバイス、サポートを得ることが出来れば、習熟が早い。

本研究では、このようなサポートを質問応答の形で行うシステムを考える。すなわち、ユーザーがシステムを使う上で生じる質問に、システム自体が対話的に答えていくことで、より早く、より深く、人間の理解を助けて習熟を可能にするシステムである。本研究ではこのように、ユーザーがシステムを使う上で生じる質問に、システム自体が対話的に答えていくことで、ユーザーの理解を促進させる事を目的としている。

本研究では、その前段階として、オセロゲームを対象とした対話的なサポートを考える。オセロのようなボードゲームも、実際にプレイしながら上達していくものである。ボードゲームでこのようなシステムを完成させることが出来れば、次のステップとしては、ボードゲームの枠組みを超えたシステムに展開出来る。

2　従来研究

従来研究では、システムをオセロゲームと質問応答システムで構成していた [1]。ユーザーがオセロゲームのプレイ中に質問が生じた場合、質問応答システムを起動し、質問をするという形であり、現在の研究も引き継ぎこの構成で行っている。

2.1　オセロプログラム

ユーザーとコンピュータが順番に打ち合うが、その際に打てる手が評価され、評価値の一番高い手を打つ。打てる手の評価には評価関数が用いられ、評価する手を打った後の局面（盤面）を評価する。評価値は、その局面の特徴に重みを掛け合わせて計算する [2]。今回考慮する特徴の一部を以下に述べる。

- 着手可能数（打てる箇所）
- 開放度（自分の石の周りの空きマスの数）の合計
- 確定石（返されない石）の数
- 打った手が危険な X 打ちになるか

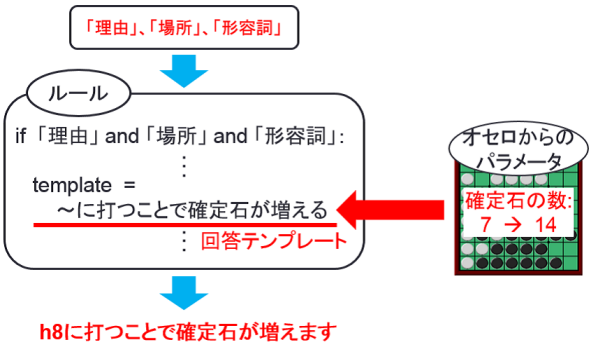

Fig. 1　ルールベースの仕組み

2.2　質問応答システム

従来研究では、質問応答システムは、ルールベースで構築しており、質問解析のために、以下に示す質問タイプ、質問キーワードを定義していた [1]。

1. 質問タイプ: 「なぜ」（質問タイプ「理由」）、「どの手」（質問タイプ「場所」）、「何」（質問タイプ「定義」）、　「どうですか？」（質問タイプ「提案」）、「どうすれば」（質問タイプ「方法」）、雑談や、挨拶、相打ち（質問タイプ「その他」）の 6 タイプ

2. 質問キーワード: 「f7」（「場所・座標」に分類）、「打つ」（「動詞」に分類）、「開放度」（「ルール・用語」に分類）、「良い」（「形容詞」に分類）など

入力文からこれらをキーワードスポッティングの形で抽出し、オセロプログラムからのパラメータと合わせて回答生成部に渡す。

回答生成部では、渡されたパラメータを条件として、それらにマッチする回答テンプレートを選択し、そこにパラメータを埋め込む形で回答を生成する (Fig.1)。

2.3　従来研究の問題点

実際に構築したシステムを使用して評価してもらったところ、様々な問題が生じた。プレイ中になされた質問を分析すると、質問や回答に多様性が求められ、if-then ルールで質問を解析、回答を生成するには限界があることが分かった。また、将来的にオセロではないドメインも扱えるようにする事を考えているので、ルールベースの場合、ドメイン毎にルールを作らなければならなくなる。以上の問題を解決するには、オセロの知識を与えるだけで、システムが自動で質問を解析し、回答出来る必要がある。

---

* A User Supporting System through Voice Question Answering. by MATSUYOSHI, Yûki, TAKIGUCHI, Tetsuya, ARIKI, Yasuo (Kobe University)

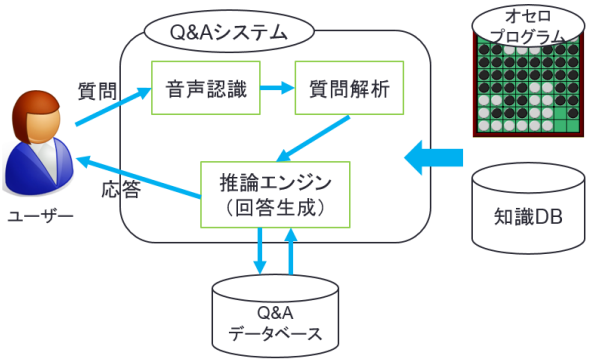

Fig. 2　システム想定図

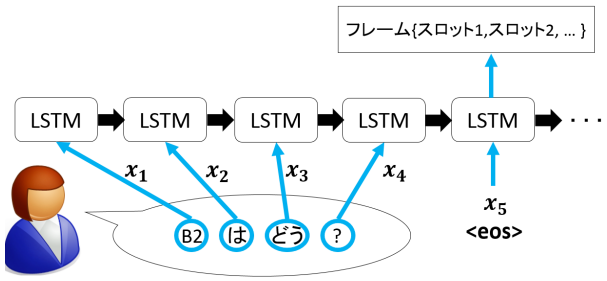

Fig. 3　モデルの概略図

3　提案システム

3.1　システム概要

従来研究の問題を解決するために、新しく構築しているシステムの想定図を Fig.2 に示す。なお、オセロプログラムについては、従来研究で用いて来たものを引き続き使用している。

ユーザーの質問が音声認識され、文字列の形で質問解析部に渡される。そして、質問解析の結果とオセロプログラムからのパラメータ、オセロの知識データベースの情報を用いて、推論エンジンにより、最適な回答を生成する。一度生成した回答はデータベースに格納しておき、同じような質問が来た際は推論エンジンを用いずにデータベースから回答する。

3.2　質問解析部

現在、質問解析部の構築に取り組んでいる。質問の解析方法として、音声認識の結果より、質問文のフレーム、スロットを推定し、抽出する。本研究では、フレームは質問の概要、ユーザーの大まかな意図を表すものとし、ユーザーの質問タイプと定義している。また、スロットは、質問中に出現するキーワードの種類（「b8」なら「場所」など）と定義している。質問タイプに関しては、従来研究で定義していた「その他」以外の質問タイプの 5 種類に加え、9 種類を追加して、合計 14 種類とした。以下に追加した質問タイプの一部を示す。

- 雑談: 挨拶や相打ちなど
- 戦略: オセロ用語や戦略についてのより深い質問

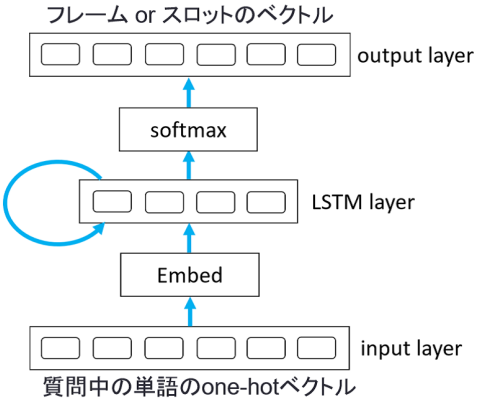

Fig. 4　LSTM ブロックの詳細

- 相手: コンピュータの手に関する質問
- 勝敗: 勝敗や形勢に関する質問

また、質問のキーワードの種類は、従来研究の定義と同じである。フレーム、スロットの抽出には、LSTM を時系列的に並べたモデルを用いた [3,4]。

モデルは Fig.3 のようになっている。まず、質問文を形態素に分割し、それぞれを one-hot な単語ベクトルに変換する。これら単語ベクトルが時系列的に LSTM ブロックに入力されていく。中間層である LSTM ブロックでは、その時点までの文脈情報を次の時刻の LSTM ブロックへ渡す。最後に <eos>（文末記号）を入力する。<eos> を入力した時刻の LSTM ブロックの出力がフレーム、スロットの推定値（フレーム、スロットのベクトル）となる。フレームのベクトルは、1 質問に対し質問タイプは 1 つなので、one-hot ベクトルになるが、スロットのベクトルは質問中にキーワードが複数あれば、その部分が「1」になるベクトルである。

LSTM ブロックの詳細は Fig.4 のようになる。単語ベクトルを質問文データに対する埋め 込み行列 (Embed) により分散表現に変換し、LSTM レイヤーに入力される。<eos> が入力される LSTM レイヤー以外は、出力層へ渡す出力は無い。<eos> が入力された LSTM レイヤーの出力が線形変換され、softmax 関数により全フレーム、全スロットに対するベクトルの出力確率が求められる。その中の最大のものから、出力するフレーム、スロットのベクトルが決まる。

4　実験

4.1　実験概要

本稿では、質問タイプだけを推定するモデルを構築し、学習を行った。学習データとしては、従来研究で構築したルールベースの質問応答システムを実際に使用してもらった際に、ユーザーの質問とその解析結果（質問タイプ、質問のキーワード）を収集してい

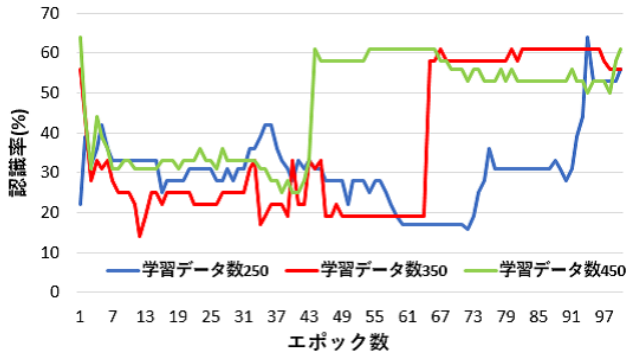

Fig. 5　学習データ数を変えることによる認識率の変化の比較

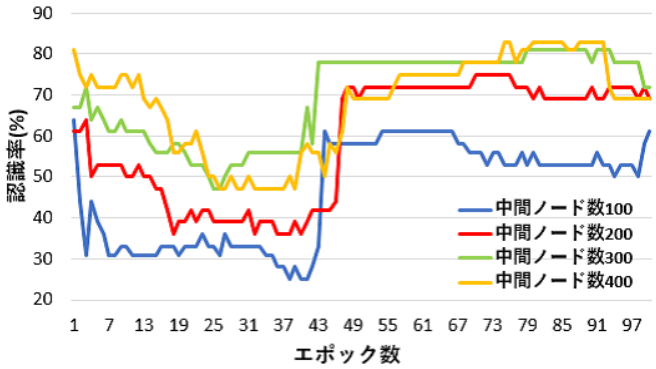

Fig. 6　中間層のノード数を変えることによる認識率の変化の比較

るので、それらを用いた。ユーザーの質問文データは全部で 487 文あり、その内の 452 文を学習データとして、残り 35 文をテストデータとし、それらに対応する質問タイプを教師データ、正解データとして用いた。

学習したモデルにテストデータ 35 文を入力し、モデルから出力された質問タイプの推定値と正解データとが一致していた数を、テストデータの文の数 (35) で割った値に 100 を掛けた値をモデルの認識率 (単位は 100%) として実験を行った。実験としては、以下の 3 つを行った。

1. 学習データ数を変えることによる認識率の変化
2. LSTM レイヤーの中間層のノード数を変えることによる認識率の変化
3. Attention の導入による認識率の変化

4.2　学習データ数を変えることによる認識率の変化

この実験では、学習に用いるデータ数を 250 文、350 文、452 文と変化させて認識率がどのように変化するのかを調べた。学習はそれぞれ 100 エポック行い、1 エポック終了する毎にモデルを出力する。100 個のモデル全てにテストデータ 35 文を入力して、認識率を求めた結果が、Fig.5 である。

Fig.5 より、学習データを増やした方がより少ないエポック数で高い認識率を出すモデルが学習出来ていることが分かる。しかし、高い認識率を出すことが出来るモデルを学習出来るという点では、どの学習データ数でもあまり差が無かった。

4.3　LSTM レイヤーの中間層のノード数を変えることによる認識率の変化

この実験では、学習に用いるデータ数は 452 文とし、LSTM レイヤーの中間層のノード数を 100、200、300、400 と変化させて認識率の変化を調べた。この際、単語ベクトルの embedding も、中間層のノード数に合わせている。4.2 と同じように、100 エポック

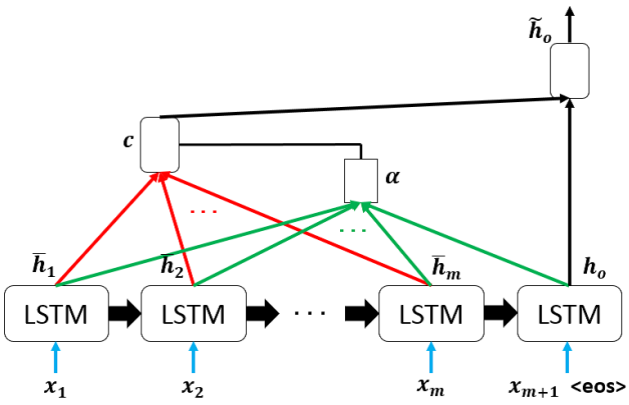

Fig. 7　Attention を導入したモデル

学習時に出力された 100 個のモデルに、テストデータを入力した結果が Fig.6 である。　Fig.6 より、中間層のノード数を増やすほど認識率が上がっていくが、ノード数が 400 の時は 300 の時に比べて全体として見ると認識率が良くなく、ノード数が 300 の方が少ないエポック数で認識率の高いモデルを学習出来ている。ノード数が 300、400 の時に、エポック数が 100 に近いモデルの認識率が落ちているが、これは、学習データの形態素数が 400 程度しかないこと、学習データ数自体が少ないことから、過学習が起きていると考えられる。

また、より認識率が高いモデルを学習出来ているのは、ノード数が 400 の時であり、83%となっているが、ノード数が 300 の時でも 81%の認識率のモデルが学習出来ており、あまり差はない。

4.4　Attention の導入による認識率の変化

Attention とは、Encoder-Decoder モデルなどでよく利用されている手法で、Encoder 側で中間層の情報を大域的に取っておいて、Decoder 側で利用するという形を取ることで、Decoder がエンコードすべき箇所を制御している。本研究では、Fig.7 の形で Attention を導入している [5,6]。 各 x_i に対して、中間層の出力である $\bar{h}_i$ を大域的に保持している。

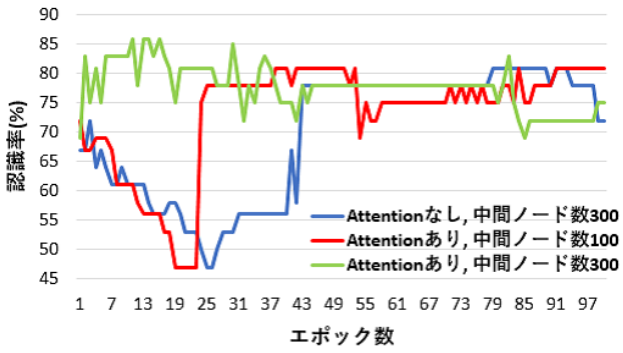

Fig. 8　Attention の導入による認識率の変化

Attention 部分について述べる。モデルの出力に対する中間層の出力を h_o とする。モデルの入力時に保持しておいた $\bar{h}_i$ を利用して、Fig.7 の $\alpha(i)$ (i=1,2,...,m) を計算する。

$$\alpha(i) = \frac{\exp(\bar{h}_i \cdot h_o)}{\sum_{j=1}^{m} \exp(\bar{h}_j \cdot h_o)} \tag{1}$$

$\alpha(i)$、$\bar{h}_i$ を用いてコンテキストベクトル c を計算する。

$$c = \sum_{i=1}^{m} \alpha(i)\bar{h}_i \tag{2}$$

c と h_o を連結したベクトル $[c; h_o]$ を作り、線形作用素 W で重みを付けて、活性化関数 $tanh$ を作用させることで、モデルの出力に対する中間層の出力 $\widetilde{h}_o$ を作る。

$$\widetilde{h}_o = tanh(W[c; h_o]) \tag{3}$$

これに対して、線形作用素で重みを付けて、softmax を通すことで、出力層の確率値を求める。

構築したモデルについて、4.2、4.3 と同じような実験を行い、Attention なしの中間層のノード数 300、Attention ありの中間層のノード数 100、300 の 3 つのモデルを比較した。結果は Fig.8 のようになった。Fig.8 を見ると、Attention なしのノード数 300 よりも、Attention ありのノード数 100 の方がより少ないエポック数で良い認識率のモデルを学習出来ていることが分かる。また、Attention ありのノード数 300 では、少ないエポック数でも十分よい認識が出来るモデルが学習出来ており、最も良いモデルの認識率は 86%にまで向上した。

以上より、Attention を導入することで、学習データが少なくても、より少ない中間層のノード数で、またより少ないエポック数で良いモデルが学習出来ることが分かった。

4.5　従来研究との比較

従来研究のルールベースのシステムと、今回の結果を比較する。ルールベースシステムの認識率は、学習データ 487 文の解析ログを見て、正しく質問タイプを解析出来ていたものの割合とする。実際に計算すると、正しく質問タイプを解析出来ていたのは 372 文であり、ルールベースシステムの認識率は 76%である。対して、今回の結果の中で一番認識率が高いモデルでは 86%であり、また、attention なしのノード数 300、400、attention ありのノード数 100、300 のモデルで、ルールベースシステムよりも高い認識率のモデルが学習出来ている。

5　おわりに

本稿では、従来研究でのルールベースの質問応答システムに変わるシステムの構築に向けて、ユーザーからの質問を LSTM を用いて解析し、質問タイプを推定するモデルの構築を行った。結果としては、学習データ数がまだまだ少なく、十分な学習が行えていないことが分かった。学習データ数をもっと増やすことが出来れば、より少ないエポック数で、またより少ない中間層のノード数で良いモデルが学習でき、学習にかかる時間を短縮出来ると考えている。

また、Attention を導入した結果、少ないエポック数、中間層のノード数でも、Attention を導入せずにノード数を増やしたモデルより認識率の高いモデルが学習でき、Attention の有効性を示すことが出来た。

本稿では質問タイプを推定するモデルのみを構築したので、今後は質問のキーワードを推定するモデルの構築に取り組みつつ、学習データ数を増やしていきたいと考えている。

謝辞　本研究の一部は、JSPS 科研費 JP17K00236 の助成を受けたものである。

参考文献

[1] 松好祐紀, 滝口哲也, 有木康雄. ”ユーザー支援を目的とした音声質問応答システム ～ オセロゲームの場合 ～,” 日本音響学会 2017 年春季研究発表会 2-P-11,2017-03

[2] Seal Software，リバーシのアルゴリズム，工学社，2003 年

[3] Chiori Hori *et al.* ”Context sensitive spoken language understanding using role dependent LSTM layers,” in Machine Learning for SLU & Interaction NIPS 2015 Workshop, 2015.

[4] Hori, T. *et al.* ”Dialog State Tracking with Attention-based Sequence-to-sequence Learning,” IEEE Workshop on Spoken Language Technology (SLT). December2016

[5] Minh-Thang Luong *et al.* ”Effective Approaches to Attention-based Neural Machine Translation,” arXiv preprint arXiv:1508.04025(2015)

[6] Dzmitry Dahdanau *et al.* ”NEURAL MACHINE TRANSLATION BY JOINTLY LEARNING TO ALIGN AND TRANSATE,” ICLR(2015)