

# ユーザに気づきを与える対話応答生成の試み

相楽 翔太\*<sup>1</sup> 呉 毘盧\*<sup>1</sup> 村上 洋平\*<sup>2</sup> 川上 皓平\*<sup>2</sup> 中辻 真\*<sup>1</sup>  
 Shota Sagara Biro Oh Yohei Murakami Kohei Kawakami Makoto Nakatsuji

NTT レゾナント\*<sup>1</sup> 日本テレビ放送網\*<sup>2</sup>  
 NTT Resonant, Inc Nippon Television Network Corporation

近年、深層学習に基づく対話応答選択に関する研究開発が目覚ましい。現状の応答選択は直前のユーザ発話に応じた短い応答を学習データに基づき選択する。本研究では、最新の対話応答技術に対し、ユーザの過去数回分の発話コンテキストを考慮し、さらに、学習データ以外に外部知識を活用する方法論を与える。それにより、対話の中で、ユーザに新たな気づきを与える応答を返すことを試みる。

## 1. はじめに

近年、深層学習を利用した対話システムが注目を集めている。しかしながら、人間同士の通常の対話ではそれまでの会話の内容が次の発話に影響を与えることが自然ではあるが、システムとユーザの対話において、過去の対話内容を明示的に学習要素とした研究は少ない。結果、過去の対話コンテキストを失い、次の応答の選択の質を劣化させている。また、人間同士で対話を行う中で、聞き手が発話者の発言への応答を行う際に、その内容を汲み取って聞き手の知識に基づき、発話者に気づきを与えることが多く行われるが、そうした対話システムの研究も少ない。結果的に現在のシステムでは、聞き手に新たな気づきを与えることは難しい。

本研究では、近年登場した深層学習に基づく質問応答手法である QA-LSTM[d] を対話応答に適用する。QA-LSTM は、発話と応答それぞれをベクトルとし表現し、学習データで対となる発話と応答のベクトル間の類似度が高くなるように、パラメータの学習を行う。本研究で提案する Context QA-LSTM は、QA-LSTM に対し以下2つの改良を行った。(1) 現在の発話ベクトルを生成する際に、過去数回分の発話者の発話ベクトルを逐次的に利用することで、過去の発話の文脈 (Context) に沿った応答を選択する。(2) 発話の内容でも重要となる単語をトラックし、応答を選択するアテンションメカニズムにおいて、さらに外部知識を利用して応答の選択候補にバイアスかける。図1に(1)の例を示す。図では、現在の時刻を  $t$  とし、 $t$  と、 $t-1$ 、 $t-2$  の3回分の過去の発話をそれぞれベクトル化し、それらをさらに pooling によってひとつの発話ベクトルとしている。その発話ベクトルと応答候補となるベクトルとの近さをコサイン類似度を用い計測する。また、図2に(2)の例を示す。この例では、ユーザが“リラックス”に関連するトピックの内容を発話したとすると、外部知識において“リラックス”と“ハーブティー”が近いということを参照し、アテンションメカニズムにて“ハーブティー”に関するトピックが応答側に入っていることを重視する動きをとる。上記2つの工夫により、会話のやりとりのコンテキストに従い、聞き手に自然と外部知識に沿った気づきを与えることを狙う。

本稿では、日本テレビの持つテレビデータを学習データとし、提案手法の評価を行った\*<sup>i</sup>。結果、提案手法が従来の QA-LSTM よりも高精度で応答を選択できる事を示した。また、被

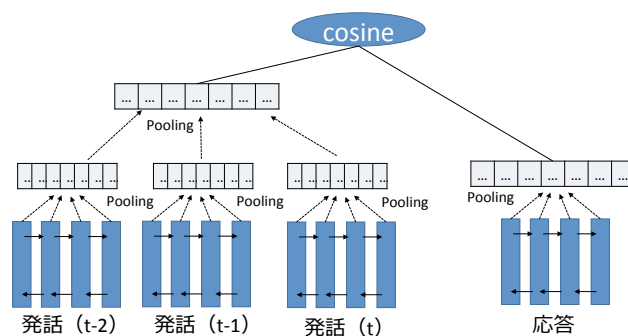


図1: Context QA-LSTM

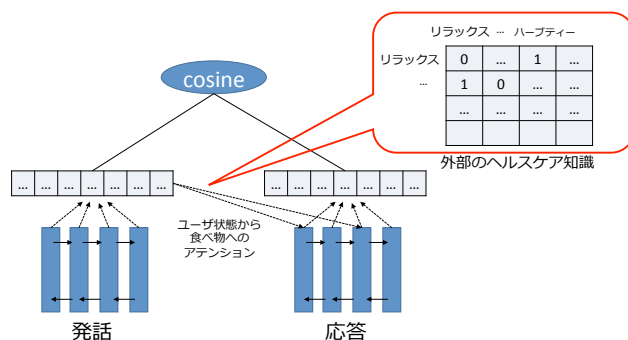


図2: 外部知識を用いたアテンション

験者による定性評価によりユーザに気づきを与えることができる可能性も示した。

## 2. 関連研究

深層学習を用い、文章のコンテキストを用いて次の文章の生成や選択を行う手法として、近年 C-LSTM[a] が提案されている。C-LSTM は、予測する文章の直前の文や段落のトピックを利用する事で、予測精度を向上させている。しかし、この手法は、対話応答ではなく、ニュース記事やコラム記事に対する文章予測を行うものである。対話応答において、コンテキストを保持する手法としては、HRED[b] がある。HRED は近年注目されている Neural Encoder-Decoder Model[c] を発展させ、Encoder および Decoder から出力されるコンテキストベクトルを逐次的に更新し、次の Encoding や Decoding に利用することで、対話のコンテキストに沿った発話生成を行っている

連絡先: 相楽 翔太, sagara@nttr.co.jp

\*i goo の AI 技術を応用!日テレとの共同実験内容を「CREATIVE TECHNOLOGY LAB」にて展示:<https://prw.kyodonews.jp/opn/release/201703069585/>

る。しかし、著者らの一次検証において、応答生成は応答選択タスクに比べて性能が劣る傾向が見られたため、本稿では、応答選択タスクである QA-LSTM フレームワーク上に提案手法を実装していく。

### 3. QA-LSTM

はじめに、LSTM について説明するとともに、本稿における用語を定義する。入力文を  $\mathbf{X} = \{\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(N)\}$  とする。ここで、 $\mathbf{x}(t)$  は  $t$  番目の単語ベクトルである。 $t$  番目の隠れベクトル  $\mathbf{h}(t)$  は以下の式で更新される。

$$\begin{aligned} \mathbf{i}_t &= \sigma(\mathbf{W}_i \mathbf{x}(t) + \mathbf{U}_i \mathbf{h}(t-1) + \mathbf{b}_i) \\ \mathbf{f}_t &= \sigma(\mathbf{W}_f \mathbf{x}(t) + \mathbf{U}_f \mathbf{h}(t-1) + \mathbf{b}_f) \\ \mathbf{o}_t &= \sigma(\mathbf{W}_o \mathbf{x}(t) + \mathbf{U}_o \mathbf{h}(t-1) + \mathbf{b}_o) \\ \tilde{\mathbf{c}}_t &= \tanh(\mathbf{W}_c \mathbf{x}(t) + \mathbf{U}_c \mathbf{h}(t-1) + \mathbf{b}_c) \\ \mathbf{c}_t &= \mathbf{i}_t * \tilde{\mathbf{c}}_t + \mathbf{f}_t * \mathbf{c}_{t-1} \\ \mathbf{h}(t) &= \mathbf{o}_t * \tanh(\mathbf{c}_t) \end{aligned}$$

LSTM ブロックは 3 つのゲート (入力ゲート  $\mathbf{i}_t$ , 忘却ゲート  $\mathbf{f}_t$ , 出力ゲート  $\mathbf{o}_t$ ) とメモリセル  $\mathbf{c}_t$  から構成される。 $\sigma$  はシグモイド関数である。 $\mathbf{W} \in R^{H \times N}$ ,  $\mathbf{U} \in R^{H \times H}$ ,  $\mathbf{b} \in R^{H \times 1}$  は本ネットワークにおける学習パラメータである。単一方向の LSTM の場合、各トークンより以前の情報を引き継いで文脈情報を学習していく一方で、各トークンの後に現れるトークンからの文脈情報を考慮することが難しい。BiLSTM は、前後の文脈情報を双方向に処理し、2 つの出力ベクトルを得る。BiLSTM ブロックの出力は、前後方向の出力 2 つを以下の式のように連結したものである  $\overrightarrow{h}(t) = \overrightarrow{h}(t) \parallel \overleftarrow{h}(t)$ 。

QA-LSTM のフレームワークは以下の通りである。まず質問  $q$ , 回答候補  $a$  からなる、与えられた入力ペア  $(q, a)$  に対し、 $q, a$  それぞれについて単語ベクトル化する。次に、それぞれ別々の BiLSTM ブロックに単語ベクトル化したシーケンスを入力として与える。BiLSTM から出力されたベクトルに対して最大プーリングを行い、双方向分の 2 つの出力ベクトルを連結することにより、 $q$  においては  $\mathbf{o}_q$ , また  $a$  においては  $\mathbf{o}_a$  として固定長の分散表現にする。最後に、入力ペア  $(q, a)$  に対してコサイン類似度  $\cos(\mathbf{o}_q, \mathbf{o}_a)$  によってスコアリングを行う。

学習要素は以下のヒンジロス関数によって定義する。

$$\mathcal{L} = \max\{0, M - \cos(\mathbf{o}_q, \mathbf{o}_a^+) + \cos(\mathbf{o}_q, \mathbf{o}_a^-)\} \quad (1)$$

ここで  $\mathbf{o}_a^+$  本来の正解の回答の出力ベクトルであり、 $\mathbf{o}_a^-$  は全回答空間からランダムに選出された不正解の回答の出力ベクトルである。また、 $M$  は損失関数におけるマージンである。複数の正解回答を持つ質問については、その質問に紐づく各回答についてそれぞれ複数の別のペアとして扱う。

さらに、QA-LSTM は、質問の内容に沿った回答選択を行うため、質問から回答へのアテンションメカニズムを導入している。これは、質問側の BiLSTM を経て、 $\mathbf{o}_q$  が計算された後、応答側の BiLSTM による  $t$  番目の出力ベクトル  $\mathbf{h}_a(t)$  に対し、アテンションメカニズムは  $\mathbf{o}_q$  に基づき応答側の出力ベクトル  $\tilde{\mathbf{h}}_a(t)$  を以下のように更新する。

$$\begin{aligned} \mathbf{m}_{a,q}(t) &= \tanh(\mathbf{W}_{am} \mathbf{h}_a(t) + \mathbf{W}_{qm} \mathbf{o}_q) \\ s_{a,q}(t) &= \exp(\mathbf{w}_{mb}^\top \mathbf{m}_{a,q}(t)) \\ \tilde{\mathbf{h}}_a(t) &= s_{a,q}(t) \cdot \mathbf{h}_a(t) \end{aligned} \quad (2)$$

ここで、 $\mathbf{W}_{am}$ ,  $\mathbf{W}_{qm}$ ,  $\mathbf{w}_{mb}$  はアテンションパラメータである。結果として、アテンションメカニズムは、発話において重要なトピックを含むワードに対し、応答側でより大きな重みを割り当てる動きであるといえる。

### 4. 提案手法

まず、Context QA-LSTM を提案し、次に、外部知識を用いたアテンションの拡張について述べる。

#### 4.1 Context QA-LSTM

まず Context QA-LSTM を支えるアイデアについて述べる。

アイデアはシンプルであり、(1) 過去のコンテキストを保持するため、過去の  $T$  回分の発話  $\mathbf{q} = \{q(1), q(2), \dots, q(T)\}$  を、現在の発話に加味すること、(2) 更に過去の発話を加味する際、過去の発話を一定の割合で弱める時間減衰を掛け、現在の発話をより重要視する、という 2 点である。

次に、提案手法の手続きについて述べる。

QA-LSTM においては、入力となる発話、応答ペアは、 $(q, a)$  であったが、Context QA-LSTM においては、過去複数発話分を加え、 $(\{q(1), \dots, q(t), \dots, q(T)\}, a)$  となる。(1) Context QA-LSTM は入力された  $T$  個の発話と、応答に対し、Word2vec 等で事前計算済みの単語ベクトルを獲得し、QA-LSTM と同様に各発話における単語の流れに対し BiLSTM を適用する。(2) そして、LSTM の順方向と逆方向それぞれに対し、各単語トークンから出力されるベクトルに対し Max-Pooling を行い、順方向の固定長のベクトル表現  $\{\vec{q}(1), \vec{q}(2), \dots, \vec{q}(T)\}$  と逆方向の固定長のベクトル表現  $\{\overleftarrow{q}(1), \dots, \overleftarrow{q}(t), \dots, \overleftarrow{q}(T)\}$  を獲得する。(3) さらに、 $T$  個の発話に対し、順方向、逆方向それぞれに対して、時間減衰をかけ、以下のようなベクトル列、 $\{\vec{q}(1) \cdot \alpha^{T-1}, \dots, \vec{q}(t) \cdot \alpha^{T-t}, \dots, \vec{q}(T)\}$ 、および  $\{\overleftarrow{q}(1) \cdot \alpha^{T-1}, \dots, \overleftarrow{q}(t) \cdot \alpha^{T-t}, \dots, \overleftarrow{q}(T)\}$  を獲得する。ここで、 $\alpha$  は時間減衰を制御するパラメータである。(4) その上で、再度、順方向、逆方向のベクトル列に対して、Max-pooling を行った上で結合することで、固定長のベクトル表現  $\mathbf{o}_q$  を獲得する。応答  $a$  に対する固定長のベクトル表現  $\mathbf{o}_a$  は QA-LSTM と同じ方法で獲得する。(5) 最後に、QA-LSTM と同様に、コサイン類似度を計算し、入力集合  $\mathbf{q}$  と応答  $a$  のスコアを計算する。Context QA-LSTM の誤差関数は、QA-LSTM と同一のものを用いる。

#### 4.2 外部知識を用いたアテンション

本稿のもう一つの提案である、外部知識を用いたアテンションについて説明をする。

まず、アイデアを述べる。先に紹介した通り、QA-LSTM では、アテンションメカニズムを導入し、過去の発話において重要となるトピックを自動判別し、応答においてそのトピックの出現性を考慮する事で、回答の選択の精度向上を行っていた。それに対し、提案手法では、応答者が専門知識 (外部知識) を持っており、その専門知識に沿って、発話者の発言に対し気づきを与えるような応答を選択できるようにすることを狙っている。そのため、発話者の発言のトピックを外部知識を用いてトピックを変換し、変換したトピックの応答の中での出現性を考慮する。

次に手続きについて述べる。

本提案では QA-LSTM のアテンションメカニズムに外部知識を入れる。本稿の外部知識は、単語間の隣接行列  $\mathbf{A}$  として表現する。つまり、 $\mathbf{A}$  の  $(i, j)$  要素  $a_{i,j}$  は、 $i$  番目の単語から

見て、 $j$  番目の単語が関連するかどうかを決定し、関連する場合に 1、関連しない場合は 0 を値として持つ。そうすると、QA-LSTMにおいて、 $\mathbf{o}_q$  が計算された後、応答側の BiLSTM による  $t$  番目の出力ベクトル  $\mathbf{h}_a(t)$  に対し、アテンションメカニズムは  $\mathbf{o}_q$  に基づき応答側の出力ベクトル  $\tilde{\mathbf{h}}_a(t)$  を以下のよう

$$\begin{aligned} \mathbf{m}_{a,q}(t) &= \tanh(\mathbf{W}_{am}\mathbf{A}^T\mathbf{h}_a(t) + \mathbf{W}_{qm}\mathbf{A}\mathbf{o}_q) \quad (3) \\ s_{a,q}(t) &= \exp(\mathbf{w}_{mb}^T\mathbf{m}_{a,q}(t)) \\ \tilde{\mathbf{h}}_a(t) &= s_{a,q}(t) \cdot \mathbf{h}_a(t) \end{aligned}$$

ここで、 $\mathbf{W}_{am}$ 、 $\mathbf{W}_{qm}$ 、 $\mathbf{w}_{mb}$  はアテンションパラメータである。結果として、提案する外部知識に基づくアテンションメカニズムは、発話における重要なトピックを含むワードに対し、外部知識のワード間の関係性を利用し、アテンションの指し示す先を明示的に変更し、変更されたトピックに関連するワードに対し応答側より大きな重みを割り当てる動きであるといえる。

これは、例えば、 $i$  番目の単語が“リラックス”だとし、 $j$  番目の単語が“ハーブティー”だとし、もし要素  $a_{i,j}$  が 1 であるとすると、“リラックス”という言葉に関するトピックが発話に含まれている場合は、“ハーブティー”に関連するトピックを含む応答が強く帰ってくるようにしたいという意図を表す。

## 5. 検証

本稿の提案する Context QA-LSTM について、結果の検証を行った。

### 5.1 データセット

今回用いたのは、日本テレビの持つ大規模なテレビデータである。また、word2vec を用いて単語ベクトルを計算する際に、上記テレビデータに加え、教えて goo の持つ約 35 万件の QA ペア、さらに、外部知識を構成するヘルスケアデータ約 300 文書を用いた。

### 5.2 検証基準とパラメータ設定

まず上記、発話応答ペアのデータセットを 10 分割し、9/10 を訓練データ、1/10 を試験データとし、検証を実施した。

検証基準としては、上位 5 件の応答予測結果に対する正解数を用いた。理由としては、応答の結果は 1 つしか用いないため、上位の精度を特に重視するためである。また、上位 5 件の結果を見ている理由は、実サービスでは応答の結果の 1 位のみを返す訳ではなく、条件に応じて上位の結果をリランキングして提示することがよくあるためである。

パラメータとしては、試行錯誤の結果、単語ベクトルの次元数を 300、発話ベクトルの次元数を 50 と設定した。また、エポック数は、200 と設定した。 $\alpha$  は 0.9 と 0.5 を比較した。

### 5.3 結果

本節で検証結果を示す。

#### 5.3.1 定量評価

結果を図 3 に示す。この結果により、提案する context QA-LSTM の方がエポックを通じ、定量的に良い結果を示すことがわかる。また、 $\alpha$  は 0.9 の時の方が概ね良い事がわかった。なお、エポック 200 までは見ていないが、 $\alpha$  を 1.0 にした時の結果は 0.9 にした時よりも悪いことを確認している。

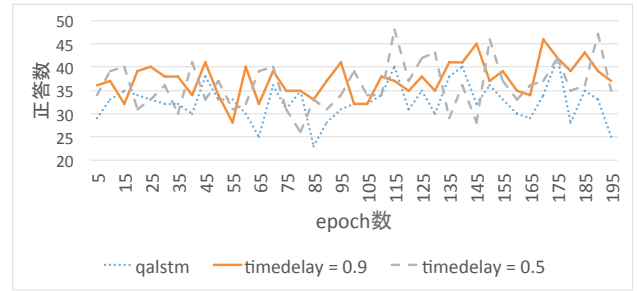


図 3: エポック数に応じた精度の比較

#### 5.3.2 定性評価

次に、QA-LSTM と、提案する context QA-LSTM(外部知識によるアテンションあり/なし) との定性評価を行った。設定とし、全手法に対し、同じ文言“いいにおいがする”から対話を試みた。結果を表 1、2、3 に示す。

表 1: QA-LSTM における対話例

ユーザ発話	システム応答
いいにおいがする	今何て言ったの？
いいにおいがするねって	冬は温かいラーメンですよ！
冬のラーメンおいしいよね	今何て言ったの？
寒いときのラーメンはおいしいよね	今何て言ったの？

表 2: 外部知識なしでの Context QA-LSTM における対話例

ユーザ発話	システム応答
いいにおいがする	すごくパンの匂い！いい匂い！
お腹すいちゃうね	うわ素敵なお店だね
買って行こう！	うん行こう！
たくさんパンがあって迷う	あすごいすごい！ヤバイ！きれい！

表 3: 外部知識有での Context QA-LSTM における対話例

ユーザ発話	システム応答
いいにおいがする	いいのか？
いいにおいだよ。でも最近便秘気味だし食べ物気をつけなとなあ	うん。それはちょっと辛いことだね。
でもお腹すいたし何か食べたいな	迷惑だったら食事に誘ったりなんかしませんよ。
じゃあ便秘に効くものとか食べに行こう。何がいいかな？	運命なんて変えてしまえばいい。一緒にパラグアイに行こう。

表 1 からわかるように、QA-LSTM では、システムの応答の流れに沿った対話が成立しづらい。一方、表 2、3 の提案手法の場合は対話が成立している。さらに、表 2 に示すとおり、外部知識なしの場合の Context QA-LSTM は、ユーザの問いかけに対して、“匂いの具体化”、“店舗の発見”、“店舗への入店の誘い”といった文脈に沿った AI 応答を返答出来ている。



図 4: Context QA-LSTM による対話例 (1)



図 5: Context QA-LSTM による対話例 (2)

さらに、外部知識有りの場合の Context QA-LSTM は、表 3 に示すように、ユーザ状態を含む会話に対して、ユーザの不調(便秘)に対して同情(辛い)する発言を選択できている。

次に、Context QA-LSTM チャットアプリとして実装し、追加検証を行ったサンプルを図 4, 図 5 に示す。

図 4 は、食物に関する対話中に話題を切り替えて対話を行った例である。話題を切り替えた直後は食物の話題を引き継いだと見られる応答を返している一方、切り替えた話題を続けるうちに切り替わった先の話題に沿った応答を返していることが分かる。対話内の話題を即時切り替えるのではなく、文脈に沿う形で少しずつ話題を切り替えていることから、過去の発言を応答の選択に用いる我々の提案手法の自然さが示されている。

図 5 は話題を統一し、初期状態(過去の発言がない状態)から対話を行った例である。本例では恋愛系統に話題を絞って対話を行った。システム側がユーザに対して好意的な応答を続けているのに対して、ユーザ側が拒否する発言を続けることで、最終的にはシステムがネガティブな応答を選択している。ユーザの各発言の文脈(過去の発言内容)に合わせて応答が自然と変更していくことが見てとれ、提案手法の有効性を示すことができている。

## 6. 結論

本稿では、ユーザの直近数回の発言も利用することでより自然な応答を選択する Context QA-LSTM と、外部知識を用

いたアテンションメカニズムの導入により対話の中でユーザへ気づきを促す 2 つの手法を提案した。そして、テレビデータを用いた検証を実施し、提案手法が従来の QA-LSTM に比べ定量的にも定性的にも、より良く次の応答を選択可能であることを示した。

## 参考文献

- [a] Shalini Ghosh, Oriol Vinyals, Brian Strope, Scott Roy, Tom Dean, and Larry Heck. Contextual LSTM (CLSTM) models for large scale NLP tasks. *CoRR*, abs/1602.06291, 2016.
- [b] Iulian Vlad Serban, Alessandro Sordoni, Yoshua Bengio, Aaron C. Courville, and Joelle Pineau. Building end-to-end dialogue systems using generative hierarchical neural network models. In *Proc. AAAI'16*, pages 3776–3784, 2016.
- [c] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. In *Proc. NIPS'14*, pages 3104–3112, 2014.
- [d] Ming Tan, Cícero Nogueira dos Santos, Bing Xiang, and Bowen Zhou. Improved representation learning for question answer matching. In *Proc. ACL'16*, pages 464–473, 2016.