The 32nd Annual Conference of the Japanese Society for Artificial Intelligence, 2018

Quasi-Recurrent Neural Networksに基づく対話モデルを用いた対話破綻検出

Dialog Breakdown Detection using Dialog Model based on Quasi-Recurrent Neural Networks

田中 涼太　　李 晃伸
Ryota Tanaka　　Akinobu Lee

名古屋工業大学 大学院情報工学専攻
Department of Computer Science and Engineering, Graduate School of Engineering, Nagoya Institute of Technology

In recent years, studies on chat-oriented dialog systems have been actively conducted due to the spread of dialog agents. On the other hand, many chat-oriented dialog systems have frequent dialog breakdown in which dialog is not smoothly performed. To tackle this problem, we propose a method to perform fast learning and robust dialog breakdown detection using Dialog Model based on Quasi-Recurrent Neural Networks (QRNN). To clarify the effectiveness, we conducted comparison experiment with other Recurrent Neural Networks (RNN) models, and show that QRNN has a faster learning and more accurate dialog breakdown detection than RNN.

1. はじめに

近年，対話エージェントの普及により，日常的な雑談を目的とする雑談対話システムの研究が盛んに行われている．一方，多くの雑談対話システムは対話が円滑に行われない現象（対話破綻）が頻発しているのが現状である．対話を行なっていてもシステムの発話が対話破綻を起こすと，会話が円滑に進まずに会話が途切れる，ユーザの対話継続意欲を阻害する，システム自体に飽きる等の可能性が考えられる．この問題を解決するためには，対話破綻の回避や対話破綻を検出してリカバリする技術が必要である．そこで，これらの技術を培うために人間とシステムが対話を行なったログに対して対話が破綻しているかを検出する対話破綻検出チャレンジ [1] が開催されている．本研究はこちらのタスクに沿って手法の提案を行う．

対話破綻検出の問題は，各システム発話に「破綻している」，「破綻とは言い切れないが，違和感を感じる」，「破綻ではない」の 3 種類のラベルが付与されており，これらのラベルを予測する問題と捉えることができる．これまでの対話破綻検出チャレンジにおいて，小林ら [2] により Neural Conversational Model (NCM)[3] を一部拡張し，対話破綻検出を行うモデルが提案された．さらに，久保ら [4] によりこの枠組みの欠点である別途学習データの収集が必要である点を改良し，高い性能を示した．

NCM の学習には一般的に Recurrent Neural Networks (RNN) が使用されるが，学習に多大な時間を要するため，高精度なモデルの構築が困難だといえる．また，久保らのモデルは NCM を用いて対話破綻検出を行なっているため，NCM の性能が対話破綻検出の精度に影響する．以上のことから，高精度な対話破綻検出を行うには，学習速度の高速化，および NCM の性能向上は必要不可欠なタスクだと考える．

本研究では Quasi-Recurrent Neural Networks (QRNN) を NCM の学習に用いる．近年の研究で，Bradbury ら [5] により RNN の 機構を Convolutional Neural Networks (CNN) によって擬似的に表現した QRNN が提案された．Bradbury らの実験で，Encoder-Decoder モデルを用いた機械翻訳タスクにおける QRNN の有効性が示唆されており，類似タスクである本タスクにおいても同様の結果が期待される．

連絡先: 田中 涼太，名古屋工業大学 大学院情報工学専攻，
rtanaka@slp.nitech.ac.jp

本稿では，2 章で対話モデルを用いた対話破綻検出について述べ，3 章で提案モデルについて述べる．4 章で評価実験について述べ，5 章で本研究のまとめを述べる．

2. 対話モデルを用いた対話破綻検出

対話破綻検出を行うことで，システム自身が発話する前に対話破綻に繋がる発話であることを認識したり，システムの発話後に自身の発話が対話破綻を起こしていることを認識することが可能である．一方で，対話破綻を事前に回避したり，リカバリを行うには，対話破綻検出を行うだけでは不十分であり，対話を行いつつ生成された発話に対して対話破綻検出を行うことで実現することができる．つまり，対話モデルと対話破綻検出を統合した枠組みでを考える必要がある．

Vinyals ら [3] により，RNN を使って発話文から応答文を生成する対話モデルとして NCM が提案された．NCM は Encoder と Decoder の 2 つのネットワークで構成されている．Encoder により発話の単語列を順々に読み込み，発話内容を表す潜在表現に圧縮し，Decoder は圧縮された表現に基づいて応答文を一単語ずつ出力する．小林ら [2] は NCM の拡張モデルとして Encoder の最終時刻における隠れ層 $\tilde{h}_T$ を Deocoder に毎時刻注入する操作を拡張したモデルを提案した．また，このモデルに発話文と応答文のペアを入力し，Encoder と Decoder の最終時刻における隠れ層 v_i，v_o を素性に分類器を作成して対話破綻検出を行った．

一方で，NCM を学習する際に，対話破綻検出チャレンジで配布されている学習データには対話が破綻しているデータも含まれるため，対話が破綻した対話とそうでない対話とが区別されないまま学習が進んでしまい正常な学習が困難である．そこで，久保ら [4] は小林らのモデルに対して，対話データに含まれる O, △, X のアノテーション分布に基づいて応答文の末尾に対話破綻ラベルを確率的に付与する手法で学習データの生成を行なった．これにより，対話破綻ラベル毎の学習が可能になり，昨年度の対話破綻検出チャレンジで最も高い F 値を示した．

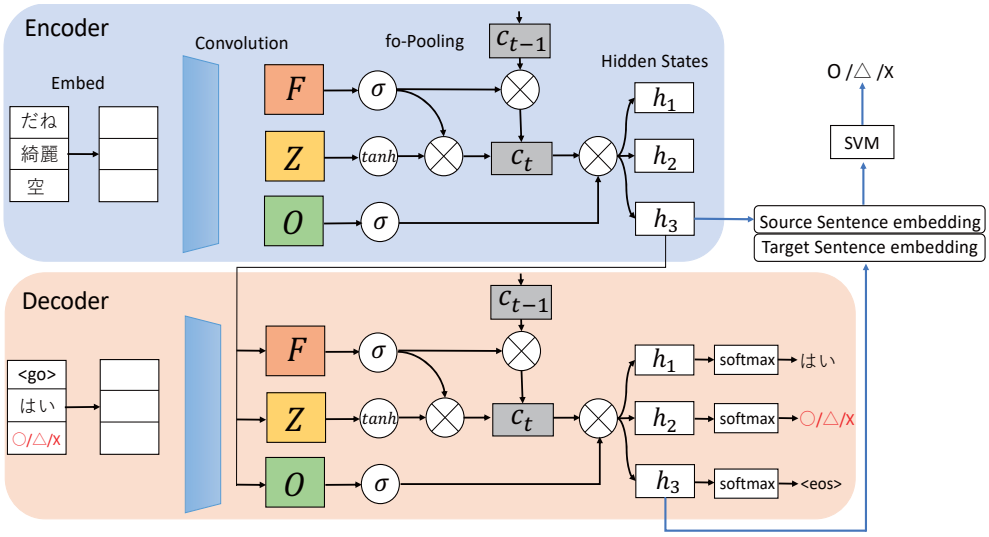

図 1: 提案モデル（久保ら [4] のモデルに対して対話モデルの学習部を QRNN に置換したモデル）

3. 提案モデル

3.1 提案モデルの概要

我々の提案するモデルは久保ら [4] のモデルをベースに対話モデルの学習を QRNN で行うモデルである．対話破綻検出の流れは以下になる．

1. O, △, X のアノテーション分布に基づいて応答文の末尾に対話破綻ラベルを付与する
2. 1. で作成した学習データを元に対話モデルを QRNN で学習する
3. 発話文と応答文のペアを対話モデルに入力し，発話文，応答文埋め込みベクトル v_i，v_o (Encoder と Decoder の最終時刻における隠れ層）を獲得する
4. v_i，v_o を素性に Support Vector Machine (SVM) で識別を行う

以上の流れを図 1 に示す．本節では，まず従来手法との差分である 2. について焦点を置き，QRNN についての導入，および QRNN に基づく対話モデルについて述べた後，3. について説明する．

3.2 Quasi-Recurrent Neural Networks

QRNN は 畳み込み層と再帰構造を持つプーリング層の 2 層により構成されるニューラルネットである [5]．入力を n 次元系，列長 T のベクトル $X \in \mathbb{R}^{T\times n}$，カーネル幅を k，カーネル数を m，ユニットへの結合重みを $\{W_z, W_f, W_o\} \in \mathbb{R}^{k\times n\times m}$ としたとき，畳み込み層では (1)~(3) 式を用いて入力から局所特徴量 $\{Z, F, O\} \in \mathbb{R}^{T\times m}$ を抽出する．ここで用いる畳み込みは未来の時刻情報を含まずに畳み込みを行うため，時系列方向に $k-1$ のパディングを入れる Masked Convolution[6] を用いる．

$$Z = \tanh(W_z * X) \quad (1)$$
$$F = \sigma(W_f * X) \quad (2)$$
$$O = \sigma(W_o * X) \quad (3)$$

プーリング層では (4)，(5) 式に基づいて過去の履歴を考慮しつつ畳み込みで得られた局所特徴量の集約 (fo-poolng) を行う．これにより，入力の各局所特徴量を独立に解釈するのではなく，文脈を考慮した大局的な解釈を行うことができる．

$$c_t = f_t \odot c_{t-1} + (1 - f_t) \odot z_t \quad (4)$$
$$h_t = o_t \odot c_t \quad (5)$$

3.3 QRNN に基づく対話モデル

(1)~(3) 式を Encoder の最終時刻における隠れ層 $\tilde{h}_T$ を用いて，Decoder では以下の式に変更したものを用いる．

$$Z = \tanh(W_z * X + V_z \tilde{h}_T) \quad (6)$$
$$F = \sigma(W_f * X + V_f \tilde{h}_T) \quad (7)$$
$$O = \sigma(W_o * X + V_o \tilde{h}_T) \quad (8)$$

3.4 発話文と応答文の特徴量抽出

QRNN で学習を行った対話モデルを発話文と応答文の特徴量を抽出する文エンコードモデルとして使用する．具体的には，対話モデルの Encoder と Decoder の最終時刻における隠れ層を特徴量として抽出する．対話モデルを QRNN で学習することで，最終時刻における隠れ層は，発話の大局情報（単語やフレーズといった局所情報を時間軸に沿って統合した情報）を表す埋め込みベクトルとしてみなすことができる．

従って，大局的に前の発話文との矛盾や情報の過不足などの齟齬が生じたとき，これらを検出することが可能である．

4. 評価実験

4.1 タスク設定

提案モデルの有効性を検証するために，学習時間の計測，および対話破綻検出を行う．学習時間の計測を行うにあたり，各モデルの層数 $\in \{1, 2, 4\}$ で変更を行い，100 epoch 学習するのに要した時間の平均を 1 epoch あたりの学習時間として比較を行う．また，対破綻検出の評価には，対話破綻検出チャレンジで用いられているラベル一致率系統と分布一致系統を使用する．具体的には，以下の評価尺度から構成される．

- Accuracy : 全ラベルの一致率．検出結果全体の正解率を測ることができる尺度．
- F-measure(X) : 破綻ラベル X の検出に関する F 値．破綻検出の正確性（Precision），網羅性 (Recall) の調和平均 (F-measure) を測ることができる尺度．
- F-measure(△+X) : ラベル △ と X を同一の破綻ラベルとみなした場合の F 値．
- Mean Squared Error (O, △, X) : 分布間の平均二乗誤差．
- JS Divergence (O, △, X) : Jensen-Shannon divergence による分布間の距離．

4.2 比較モデル

評価実験では以下の 3 種類のモデルを比較する．

GRU (従来手法) 対話モデルの学習を GRU で行う

LSTM (従来手法) 対話モデルの学習を LSTM で行う

QRNN (提案手法) 対話モデルの学習を QRNN で行う

以上の全てのモデルを Chainer[*1] を用いて実装した．実験に用いた計算機の GPU には，GeForce GTX Titan X ，GPU 演算のためのツールには CUDA-8.0[*2] を用いた．

4.3 データセット

本実験では対話破綻検出チャレンジ 1，および対話破綻検出チャレンジ 2 で用意されている配布データを使用した．配布データに含まれる対話システムは DCM，DIT，IRS から構成される．様々な形態の対話システムのデータを扱うことで，汎用的な対話破綻検出器の構築が期待できる．また，対話データには以下のラベルのアノテートが行われている．

- **O 破綻ではない :** 問題なく対話が継続できる
- **△ 破綻とは言い切れないが，違和感を感じる :** スムーズな対話継続が困難
- **X 破綻している :** 対話を継続することが困難

表 1 に配布データの詳細を示す．ただし，1 対話は 21 個の発話で構成されている．

表 1: 対話破綻検出チャレンジの配布データ

システム	データ名	対話数	O	△	X (%)
DCM	rest1046	1046	58.3	25.3	16.4
	DBDC dev/test	20/80	37.1	32.2	30.6
	DCM dev/test	50/50	39.8	30.2	29.9
DIT	DIT dev/test	50/50	33.0	27.4	39.5
IRS	IRS dev/test	50/50	37.4	24.3	38.3

対話データには最大 30 人の評価者が破綻ラベル (O, △, X) の付与を行っているため，最も多く付与されているラベルを正解ラベルとした．また，ラベルの数が同数の場合は O, △, X の順の優先度で正解ラベルを決定した．前処理として，全ての文に対して Mecab[*3] を用いて形態素解析を行った．Mecab の辞書には多くの固有名詞に対応している ipadic-negologd[*4] を使用した．語彙は単語出現頻度上位 4000 語に制限し，語彙に含まれない単語を未知語 <unk> に置換した．

モデル選択を行うために DCM dev，DIT dev，IRS dev のうち 60 対話を開発データとして利用した．対話モデルの学習には，rest1046，DBDC dev/test と DCM dev，DIT dev，IRS dev のうち開発データに含まれないデータを使用した．対話破綻検出器の学習には，対話モデルの学習で用いた学習データのうち，ラベルの偏りが大きいデータ (rest1046) を除くデータを使用した．評価データは DCM test，DIT test，IRS test を使用した．

4.4 学習条件

入力層を 256 次元，中間層には 256 次元の RNN (LSTM, GRU)，QRNN ユニットをどちらか一方を 2 層重ねた．QRNN の畳み込みにおける入力チャネル数，出力チャネル数，カーネル幅，ストライド幅を表 2 に示す．

表 2: QRNN の畳み込みパラメータ

layer	in-ch	out-ch	kernel	stride
1st (Encoder)	256	256 × 3	(6,1)	1
2nd (Encoder)	256	256 × 3	(2,1)	1
1st (Decoder)	256	256 × 3	(1,1)	1
2nd (Decoder)	256	256 × 3	(1,1)	1

全ての結合重みは，平均 0，分散 $\frac{1}{\sqrt{256}}$ のガウス分布に基づいて初期化を行った．学習はミニバッチ学習で行い，ミニバッチサイズを 32 とした．ミニバッチごとの勾配爆発を防ぐために最大ノルムを 5 として勾配クリッピングを行った．目的関数には予測単語ごとの交差エントロピー誤差を使用し，Adam (alpha=0.01，beta1=0.9，beta2=0.999，eps=1×10^{-8}) で最適化を行なった．また，200 epoch の学習を行う過程において 20 epoch 終了するたびに rbf カーネルを用いた SVM で学習を行い，開発データを用いて F-measure の算出した．ここで最も高い性能を示したモデルを採用した．過学習を防ぐ目的に，中間層と出力層に dropout ，目的関数に weight decay を導入した．dropout$\in \{0.3, 0.4, 0.5\}$，weight decay$\in \{2\times10^{-3}, 2\times10^{-4}, 2\times10^{-5}\}$ の値でパラメータの変更を行い，最も F-measure の高いパラメータを採用した結果，GRU の dropout は 0.3，weight decay は 2×10^{-4}，LSTM の dropout は 0.5，weight decay は 2×10^{-5}，QRNN の dropout は 0.3，weight decay は 2×10^{-4} となった．学習時間の計測の際は，dropout による学習時間の影響が考えられるため，dropout の適用は行わないものとする．

4.5 実験結果

層数を変動させた時の各モデルの学習速度の計測結果を表 3 に示す．結果は QRNN が最も速く，最も遅い LSTM と比較すると，約 3.2 倍の速度で学習が行えることが確認できる．これは，QRNN が入力に対して並列処理を行えることによる影響だと考えられる．

各モデルの対話破綻検出の結果を表 4 に示す．DCM の Accuracy と F-measure(△+X) を除いた指標において QRNN が RNN よりも高精度に対話破綻を検出することが確認でき

∗1 http://chainer.org
∗2 https://developer.nvidia.com/cuda-toolkit
∗3 http://mecab.sourceforge.net
∗4 https://github.com/neologd/mecab-ipadic-neologd

表 3: 学習速度の計測結果 [epoch/s]

Model \ Layer	1	2	4
GRU	59.11	107.7	210.6
LSTM	71.80	139.2	254.4
QRNN	**22.51**	**43.93**	**85.26**

表 4: 対話破綻検出の結果

System	Model	Accuracy	F-measure (X)	F-measure (△+X)	MSE (O,△,X)	JSD (O,△,X)
DCM	GRU	0.4709	0.4872	0.6175	0.0657	0.1309
	LSTM	**0.5018**	0.4510	**0.6414**	0.0651	0.1270
	QRNN	0.4909	**0.5116**	0.5880	**0.0635**	**0.1256**
DIT	GRU	0.5873	0.6853	0.7995	0.0463	0.0919
	LSTM	0.6073	0.7131	0.8973	0.0445	0.0874
	QRNN	**0.6091**	**0.7137**	**0.9175**	**0.0367**	**0.0762**
IRS	GRU	0.4873	0.5424	0.6944	0.0773	0.1485
	LSTM	0.5091	0.5817	0.7207	0.0753	0.1439
	QRNN	**0.5291**	**0.6314**	**0.7970**	**0.0687**	**0.1350**

る．RNN では誤った検出を行なっていたが，QRNN で正しく検出が行えた対話例を以下に示す．

ユーザ： 今日も暑いですね

システム： 大通会場と JR 東日本なら，どちらに関心がありますか？

ユーザ： 何のことですか？

システム： シンガポールと奈良なら，どちらが楽しいでしょうか？

「何のことですか？」という発話に対して，「シンガポールと奈良なら、どちらが楽しいでしょうか？」と応答しているように，本来は質問に答える対話行為や話題を追従する応答をするべきであるが疑問文で返していたり，全く無関係な話題に転換している．このような対話に対して QRNN は対話行為や話題といった対話の特徴を捉えることができ，正しく検出できたのだと考えられる．

5. おわりに

本研究では，高速かつ高精度な対話破綻検出器の構築を行うために，対話モデルの学習に QRNN を適用し，対話モデルを用いた対話破綻検出を提案した．ベースラインモデルとして，対話モデルの学習に RNN (LSTM，GRU) を用いた対話破綻検出の手法を比較対象とした評価実験を行なった．

評価実験では対話モデルの学習に QRNN を適用することで，入力系列に対する並列処理が可能になり，学習速度が約 3.2 倍の高速化を実現した．また，文脈に沿った大局的な学習が可能になったことで，不自然な対話行為や無関係な話題の転換といった検出が可能になり，検出精度の改善を示した．

今後の課題として，対話破綻検出器の素性の最適化，対話モデルに用いる学習データに大規模なデータセットを用いることで，より高精度な対話破綻検出が可能になると考えられる．また，本研究で作成した枠組みを用いることで，対話破綻検出の結果を発話生成に生かす手法のアプローチになると考えられる．

参考文献

[1] 東中竜一郎，船越孝太郎，稲葉通将，荒瀬由紀，角森唯子，“対話破綻検出チャレ ンジ 2，”第 78 回言語・音声理解と対話処理研究会 (第 7 回対話システムシンポ ジウム)，人工知能学会研究会資料 SIG-SLUD-B5025-19，pp.64-69，2016.

[2] 小林颯介，海野裕也，福田昌昭，“再帰型ニューラルネットワークを用いた対話破綻 検出と言語モデルのマルチタスク学習， ”第 75 回言語・音声理解と対話処理研究会 (第 6 回対話システムシンポジウム)，人工知能学会研究会資料 SIG-SLUD-075-09， pp.41-46，2015.

[3] Oriol Vinyals，Quoc Le，“ A Neural Conversational Model，” arXiv:1506.05869，2015.

[4] 久保隆宏，中山光樹，“Neural Conversational Model を用いた対話と破綻の同時 学習，”第 78 回言語・音声理解と対話処理研究会 (第 7 回対話システムシンポジ ウム)，人工知能学会研究会資料 SIG-SLUD-B5025-19，pp.94-97，2016.

[5] James Bradbury，Stephen Merity，Caiming Xiong，Richard Socher，“Quasi-Recurrent Neural Networks，” arXiv:1611.01576，2016.

[6] Aaron van den Oord，Nal Kalchbrenner，Koray Kavukcuoglu. “ Pixel recurrent neural networks，” arXiv:1601.06759，2016.