RNNエンコーダによる文脈を考慮した対話破綻検出

Dialogue Breakdown Detection using RNN Encoders

稲葉 通将 [1*]　高橋 健一 [1]
Michimasa INABA[1]　Kenichi TAKAHASHI[1]

[1] 広島市立大学大学院情報科学研究科
[1] Graduate School of Information Sciences, Hiroshima City University

Abstract: Thip paper describes a method for dialogue breakdown detection using recurrent neural network encoders. Specifically, the proposed method processes word sequences in utterances and utterance sequences in context via recurrent neural network encoders. Experimenal results show that the proposed methods outperform the baseline method in estimation of annotation distribution and detection of ×. However, in detection of △ and ×, the performances of our methods are lower then the baseline method.

1　はじめに

本稿では，対話における 2 つの系列，すなわち単語の系列 (発話) と発話の系列 (文脈) を Recurrent Neural Network(RNN) エンコーダにより処理することで対話破綻検出を行う手法について述べる．提案する破綻検出手法で用いる RNN の構造は Neural Utterance Ranking モデル [1] をもとにしたものである．また，独自のデータとして，我々が作成した対話システムを用いて収集した対話データを公式に配布されているデータとともに学習データとして用いる．

なお，対話破綻検出チャレンジ，および配布されているデータの詳細については文献 [2] を参照されたい．

2　対話破綻検出手法

対話破綻検出チャレンジで提供されている対話データでは，全ての対話システムの発話に対し，対話破綻のアノテーションが行われている．アノテーションは ○・△・×の 3 分類で行われており，それぞれ「破綻ではない」,「破綻とは言い切れないが違和感を感じる」,「破綻」を意味する．データには複数のアノテータ (2〜30 名) が個別に付与したアノテーションが統合されること無くそのまま収録されており，本論文で提案する破綻検出手法は，アノテータが○・△・×のそれぞれに対し，どのような割合でアノテーションを行ったかという分布を推定する．

*連絡先： 広島市立大学大学院情報科学研究科
〒 731-3194　広島市安佐南区大塚東 3-4-1
E-mail: inaba@hiroshima-cu.ac.jp

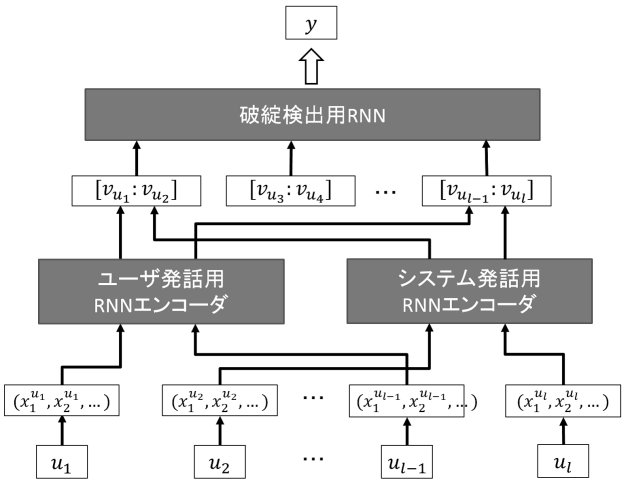

図 1: 提案手法

2.1　発話のエンコード

提案手法で用いる RNN の概要を図 1 に示す．提案手法では，破綻検出対象となる対話システムの発話，およびそれ以前の文脈中の全ての発話をそれぞれ RNN エンコーダを用いて固定長のベクトルへエンコードする．まず，Mecab[3] により発話を単語に分割し，単語の系列 $w = (w_1, w_2, \ldots, w_n)$ を得る．次に，単語の系列 w を単語の分散表現の系列 $x = (x_1, x_2, \ldots, x_n)$ に変換し，この系列を RNN への入力とする．分散表現への変換は Mikolov らの手法 [4] を実装した word2vec を用いる．

RNN エンコーダは順方向と逆方向の 2 種類を用意する．まず，順方向 RNN は x を文頭から順に入力として

受け取り，各入力に対応する $\overrightarrow{h} = (\overrightarrow{h_1}, \overrightarrow{h_2}, \ldots, \overrightarrow{h_n})$ を出力する．逆方向 RNN には x を最後から逆順に入力し，$\overleftarrow{h} = (\overleftarrow{h_1}, \overleftarrow{h_2}, \ldots, \overleftarrow{h_n})$ を出力する．最終的に獲得する発話ベクトル v は順方向と逆方向の RNN のそれぞれの最後の出力を連結した $v = [\overrightarrow{h_n}; \overleftarrow{h_n}]$ となる (; はベクトルの連結を意味する)．

また，発話の順位付けのために抽出が必要な情報はシステムの発話とユーザの発話では異なると考えられる．そこで，発話のエンコードはシステム発話・発話候補とユーザ発話で別の RNN を使用する．

2.2 破綻ラベルの分布推定

破綻検出には，発話のエンコードに用いた RNN とは別の破綻検出用 RNN を用いる． この破綻検出用用の RNN は入力側から順に，LSTM の層を 2 層，ReLu を活性化関数に用いた通常のニューラルネットの中間層を 2 層，最後に出力層という構造を持つ．与えられた文脈に対するシステム発話の破綻を検出するためには，文脈最後のユーザの発話と検出対象のシステム発話のみから判断するのではなく，それよりも前の文脈における，システムとユーザのやり取りの系列も考慮する必要がある．そこで，この RNN には，文脈と破綻検出対象の発話から構成したベクトル系列を時系列順に入力することで，3 次元 (破綻ラベル○・△・×に対応) の確率分布 y を得る．

入力ベクトル系列 v^c は，発話ベクトル系列 $v_c = (v_{u_1}, v_{u_2}, \ldots, v_{u_{l-1}}, v_{u_l})$ を用いて以下のように構成する．

$$v^c = ([v_{u_1}; v_{u_2}], [v_{u_3}; v_{u_4}] \ldots, [v_{u_{l-1}}; v_{u_l}])$$

このように，RNN にユーザの発話とそれに対するシステムの発話をペアで入力することで，検出対象のシステム発話とその直前のユーザ発話の関連性を捉えることに加え，それよりも前のユーザとシステムのやり取りを考慮することが可能となる．

損失関数には正解分布との間の Mean squared error(MSE) を用い，AdaGrad[5] により学習を行う．

2.3 例外的な処理

対話破綻検出チャレンジにおける対話データでは，対話システムの発話から対話を開始する．したがって，対話の最初のシステムの発話で対話が破綻することは少ない．そこで提案手法では，例外的に対話における最初のシステム発話の分布は $(○, △, ×) = (1.0, 0.0, 0.0)$ とする．

3 評価実験

3.1 実験設定

提案した破綻検出手法の性能評価のため，実験を行った．実験に使用した対話データを表 1 に示した．表中の学習，モデル選択，評価はそれぞれ学習に使用したデータ数，モデル選択に使用したデータ数，評価に使用したデータ数を意味する．init100，rest1046，DBDC 開発・評価用は 2015 年の対話破綻検出チャレンジで配布されたデータであり，DCM，IRS，DIT は今回のチャレンジで配布されたデータである．NUR は我々が開発した対話システムを用いて独自に収集を行ったデータである．NUR の収集のために使用した対話システムは，Twitter を用いた発話候補文獲得手法 [6] により獲得した発話候補から，Neural Utterance Ranking モデル [1] により文脈に応じて適切な発話を選択することで対話を進めるシステムである．本データは，対話ルール，アノテーションルール等は本チャレンジに準拠し，収集した．

学習はモデルのパラメータの初期値を変更し 4 回行い，1 epoch ごとに 80 個のモデル選択用のデータで性能を評価する．その結果を元に，評価実験は以下の run1〜3 の 3 つの設定で行う．

run1
: モデル選択用のデータで MSE が最小となったモデル．

run2
: モデル選択用のデータでの検出において F 値が最大となったモデル．

run3
: 4 回の学習のそれぞれにおいて，モデル選択用のデータで MSE が最小となったモデルを使用し，それぞれの出力の平均を使用．

また本実験では，ベースラインとして 2015 年の対話破綻検出チャレンジで配布された条件付き確率場を用いた検出手法との比較も実施する．ベースライン手法では学習データとして提案手法と同じ 1436 個のデータを使用し，学習時のしきい値は $t = 0$ とした．

3.2 実験結果

実験結果を表 2〜表 6 に示す．表 2，表 3，表 4 はそれぞれ分布一致系統における DCM，IRS，DIT の結果であり，表 5 は DCM，IRS，DIT のデータをまとめて集計したものである．また，表 6 はラベル一致系統における DCM，IRS，DIT のデータをまとめて集計した結果である．

表 1: 使用データ

データ名	アノテータ人数	学習	モデル選択	評価	対話数
init100	24	100	0	0	100
rest1046	2〜3	1046	0	0	1046
DBDC 開発・評価用	20	100	0	0	100
DCM	30	30	20	50	100
IRS	30	30	20	50	100
DIT	30	30	20	50	100
NUR	34	100	20	0	120
合計	-	1436	80	150	1666

表 2: 分布一致系統結果 (DCM)

	JSD	JSD (T+X)	JSD (O+T)	MSE	MSE (T+X)	MSE (O+T)
baseline	0.407	0.224	0.244	0.221	0.194	0.215
run1	0.101	0.073	0.062	0.056	0.075	0.067
run2	0.118	0.078	0.073	0.068	0.084	0.078
run3	**0.100**	**0.072**	**0.061**	**0.055**	**0.074**	**0.067**

表 3: 分布一致系統結果 (IRS)

	JSD	JSD (T+X)	JSD (O+T)	MSE	MSE (T+X)	MSE (O+T)
baseline	0.433	0.243	0.323	0.239	0.218	0.301
run1	**0.118**	**0.080**	**0.082**	**0.065**	**0.079**	0.091
run2	0.156	0.100	0.114	0.091	0.102	0.128
run3	0.118	0.081	0.082	0.066	0.080	**0.091**

表 4: 分布一致系統結果 (DIT)

	JSD	JSD (T+X)	JSD (O+T)	MSE	MSE (T+X)	MSE (O+T)
baseline	0.366	0.166	0.313	0.183	0.125	0.281
run1	0.055	0.036	0.037	0.031	0.035	0.044
run2	0.078	0.045	0.056	0.045	0.045	0.065
run3	**0.052**	**0.033**	**0.035**	**0.029**	**0.032**	**0.041**

表 5: 分布一致系統結果 (総合)

	JSD	JSD (T+X)	JSD (O+T)	MSE	MSE (T+X)	MSE (O+T)
baseline	0.402	0.211	0.293	0.214	0.179	0.266
run1	0.091	0.063	0.061	0.051	0.063	0.067
run2	0.118	0.075	0.081	0.068	0.077	0.090
run3	**0.090**	**0.062**	**0.060**	**0.050**	**0.062**	**0.066**

分布一致系統の評価では IRS を除き，run3 が最も良い結果となった．また，IRS においても run1 との差は僅かであることが確認できる．したがって，複数の学習済みモデルの平均を用いる手法は，概ね有効であったと考えられる．

一方，ラベル一致系統の評価では，モデル選択用の

表 6: ラベル一致系統結果 (総合)

	一致率	精度 (X)	再現率 (X)	F 値 (X)	精度 (T+X)	再現率 (T+X)	F 値 (T+X)
baseline	0.490	0.533	0.603	0.560	0.789	**0.846**	**0.817**
run1	**0.550**	**0.597**	**0.620**	**0.608**	**0.860**	0.551	0.672
run2	0.512	0.574	0.590	0.582	0.824	0.766	0.794
run3	0.544	0.590	0.597	0.594	0.854	0.624	0.621

データで F 値が最大のモデルを使用した run2 よりも，run1 の方が良い性能を示した．また，F 値 (X) では提案手法はルールベースを上回ったものの，F 値 (T+X) ではすべての run がルールベースに劣る結果となった．

4 まとめ

本稿では，Recurrent Neural Network(RNN) エンコーダを用いた対話破綻検出手法について述べた．提案手法では，文脈中の発話を RNN によりエンコードし，その結果得られた発話ベクトルをさらに時系列順に別の RNN でエンコードすることで，対話破綻アノテーションの分布を推定した．各発話は形態素解析と word2vec を用いて単語の分散表現の系列に変換し，この系列を発話用の RNN エンコーダの入力とした．

実験では 3 種類の設定で提案手法の評価を行った．分布一致系統の評価では，複数の学習済みモデルの平均を用いた設定が最も良い結果となった．一方，ラベル一致系統の評価では，モデル選択時に分布一致系統で最も良い結果となったモデルを用いた設定が良い結果となり，意図したものとは異なる結果となった．この点に関することを含め，今後は結果の詳細な分析を行う予定である．

参考文献

[1] Michimasa Inaba and Kenichi Takahashi. Neural utterance ranking model for conversational dialogue systems. In *Proc. SIGDIAL2016*, pp. 1–9, 2016.

[2] 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 7 回対話システムシンポジウム, 2016.

[3] T. Kudo. Mecab: Yet another part-of-speech and morphological analyzer. *http://taku910.github.io/mecab/*, 2005.

[4] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pp. 3111–3119, 2013.

[5] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *The Journal of Machine Learning Research*, Vol. 12, pp. 2121–2159, 2011.

[6] 稲葉通将, 神園彩香, 高橋健一. Twitter を用いた非タスク指向型対話システムのための発話候補文獲得. 人工知能学会論文誌, Vol. 29, No. 1, pp. 21–31, 2014.