人工知能学会研究会資料
SIG-SLUD-B502-13

Long Short-Term Memory Recurrent Neural Network を用いた対話破綻検出

Dialogue Breakdown Detection using Long Short-Term Memory Recurrent Neural Network

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 with long short-term memory cells (LSTM-RNN). The proposed method uses a pair of system's utterance and preceding user's utterance for dialogue breakdown detection. Each utterances are converted into sequences of vector representation of word by word2vec and we use it for the input of the LSTM-RNN. In our model, we build two LSTM-RNNs, for processing user's utterances and system's uttearnces. The sequences of user's utterance and system's utterance are processed by each LSTM-RNNs and our model estimates distributions of annotations of dialogue breakdown by integrating each outputs. Experimenal results show that the proposed methods outperform the baseline method in detection of X and estimation of annotation distribution. However, in detection of △ and X, the performances of our methods are lower then the baseline method.

1　はじめに

近年，非タスク指向型対話システムの研究は活発化している．また，マイクロソフトの女子高生 AI りんな [1] やリクルートのパン田一郎 [2] など，様々な企業が対話システムを公開するなど，産業界における活用も進んでいる．しかし，その性能は未だ発展途上であり，対話の途中で話が破綻してしまうことも多い．一方，対話が破綻する可能性を事前に推定できれば，それを回避できる可能性が高まる [1] など，よりよい対話システムの実現に有用な技術となる．

本論文では，Long short-term memory を中間層に用いた Recurrent Neural Network(LSTM-RNN) による対話破綻検出手法について述べる．実験では，対話破綻検出チャレンジで提供されているデータセットを用いて学習・評価を行う．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．

*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
[1]http://rinna.jp/rinna/
[2]http://line.froma.com/

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．

提案手法では，破綻検出対象となる対話システムの発話と，その直前のユーザ発話の 2 発話のみを用いる．各発話は Mecab[3] を用いて単語に分割し，単語の系列を得る．次に単語の系列を単語の分散表現の系列に変換し，この系列を LSTM-RNN 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．LSTM-RNN はシステム発話用とユーザ発話用の 2 種類を用意する．システム発話から得た分散表現系列と，ユーザ発話から得た分散表現系列をそれぞれ入力し，2 つの出力を統合して分布の推定結果を得る．

2.1 Long Short-Term Memory Recurrent Neural Network

Recurrent Neural Network(RNN) は系列データを扱うためのモデルであり，前時刻の中間層を現時刻の入力としても用いることで，内部状態を保持しながら学習を行うことができる．しかし，通常の RNN は逆誤差伝播による学習を行う際，勾配が減衰するという問題 (勾配消失) が存在する．

Long short-term memory(LSTM)[5] は勾配消失の問題を解決するために提案されたユニットの 1 つである．LSTM は Constant Error Carousel(CEC) と呼ばれる記憶素子にエラーを選択的に取り込み，保持することで勾配の消失を防ぐ．LSTM は，入力ゲート，忘却ゲート，出力ゲートの 3 つのゲートを持ち，どのようなときに CEC にエラーを取り込み，消去し，出力するかを制御する．v_t を時刻 t における単語の分散表現，h_t を時刻 t における出力とすると，LSTM は以下の式で表せる．

$$i_t = \sigma(W_i v_t + U_i h_{t-1} + b_i)$$

$$f_t = \sigma(W_f v_t + U_f h_{t-1} + b_f)$$

$$o_t = \sigma(W_o v_t + U_o h_{t-1} + V_o c_t + b_f)$$

$$c_t = i_t \odot \tanh(W_c v_t + U_c h_{t-1} + b_c) + f_t \odot c_{t-1}$$

$$h_t = o_t \odot \tanh(c_t)$$

式中の σ はシグモイド関数であり，i_t, f_t, o_t はそれぞれ入力ゲート，忘却ゲート，出力ゲート，c_t は CEC である．また，$\odot$ はベクトルの要素ごとの積を意味する．

2.2 提案モデル

対話が破綻するケースは様々であるが，生成した発話文が文法的に誤っており，意味不明の発話が出力されてしまうというように，ユーザ発話の内容に依存せず破綻が発生する場合も多い．また，システムの過去の発言に関する質問など，システムが適切に応答するのが難しいユーザ発話も存在する．そこで提案手法では，ユーザ発話用，システム発話用の 2 つの LSTM-RNN を用いる．それぞれがユーザ発話・システム発話を個別に学習することで，破綻検出につながる情報を効率よく処理可能となることが期待できる．

各 RNN は入力系列の最後の要素が読み込まれた時点で固定長のベクトルを出力する．ユーザ発話用 RNN の出力を o_t^u，システム発話用の出力を $o_{t'}^s$ とすると，アノテーションの分布 y は以下の式により求める．

$$y = softmax(W_u o_t^u + W_s o_{t'}^s + b_{us}) \qquad (1)$$

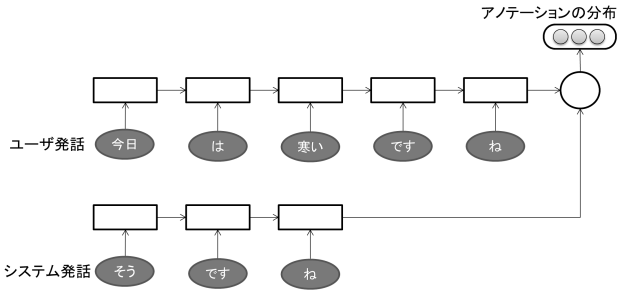

図 1: LSTM-RNN による対話破綻検出の例

損失関数には，正解の分布との間の Mean squared error を用いる．

表 1 に提案する LSTM-RNN を用いた破綻検出の例を示す．ユーザ発話とシステム発話はそれぞれ別の LSTM-RNN に順に入力される．それぞれの出力は式 (1) により統合され，最終的に 3 次元のアノテーションの分布を得る．

実験では，3 種類の構成・設定の異なる LSTM-RNN を用いて破綻検出を行った．以下ではそれぞれについて述べる．

2 層の LSTM(2-LSTM)

中間層に LSTM の層を 2 つ重ねた RNN を用いて破綻検出を行う．分散表現，および 1 層あたりの LSTM の次元数は 1000 とする．

2 層の LSTM ＋終端記号 (2-LSTM + TS)

分散表現の系列の末尾に終端記号を付与し，終端記号が入力された時点の出力を LSTM-RNN の出力として用いる．LSTM-RNN が出力を行う時点を終端記号により明示することで，LSTM の出力ゲートが破綻検出のために適切に動作することを期待し，設定した．分散表現，および LSTM の CEC の次元数は 1001 とする．分散表現の 1001 次元目の要素は終端記号か否かを示すものであり，終端記号以外は 0 となる．終端記号は 0〜1000 次元目までが 0，1001 次元目が 1 のベクトルを用いる．それ以外の設定は 2 層の LSTM と同一である．

Bidirectional LSTM(BLSTM)

時刻 $t-1$ の隠れ状態を t の隠れ状態の入力として用いる順方向の LSTM に加え，時刻 $t+1$ の隠れ状態を t の隠れ状態の入力として用いる逆方向の LSTM を中間層に用いる Bidirectional LSTM[6] を 2 つ重ねた RNN を用いて破綻検出を行う．分散表現の次元数は 1000，Bidirectional

LSTM の次元数は 2000(順方向と逆方向で 1000 次元ずつ) 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，対話破綻検出チャレンジで配布されているデータを用いて実験を行う．

実験に使用した対話データには，全てのシステム発話にアノテーションされているが，データによってアノテータの人数が異なる．配布されているデータの内訳はアノテータが 24 名のものが 100 個，2〜3 名のものが 1046 個，30 名のものが 100 個である．本実験では，アノテータが 24 名のデータ 50 個をモデル選択のため使用する．学習中に 50 個のデータで繰り返し性能を評価し，最も性能の良かった時点のパラメータを評価用に用いる．30 名のデータ 80 個は評価データとして使用し，それ以外のデータを学習データとして用いる．

また，本論文で提案した 3 つのモデルのほか，ベースラインとして対話破綻検出チャレンジで配布されている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，アノテータが 30 名のデータ 80 個を評価データとして使用し，それ以外のデータを学習データとして用いる．ベースライン手法の学習時のしきい値は 0.1 とした．

評価はアノテーション×，および△と×の検出性能，およびモデルの出力した各アノテーションの分布と評価データ (正解) の分布の間の Jensen-Shannon divergence と Mean squared error により評価する．正解のアノテーションは分布中で最大の割合を持つアノテーションとする．ただし，アノテータ間で評価が分かれたアノテーションの扱いを調整するため，しきい値 t を用いる．△と×に関しては，割合が t 以上かつ最大である場合に正解のアノテーションとし，割合が最大であっても，t 未満であればアノテーションは○とする．検出精度の評価では，t を 0.1 から 1.0 まで 0.1 刻みで変化させて評価する．

3.2 実験結果

実験結果を図 2，図 3，表 1 に示す．図 2 は×を破綻，図 3 は△と×を破綻とした場合の F 値であり，表 1 は分布間の JensenShannon divergence(JSD) と Mean squared error(MSE) である．表 1 における (T+X) は△と×を，(O+T) は○と△をそれぞれ同じアノテーションとみなした場合の結果である．

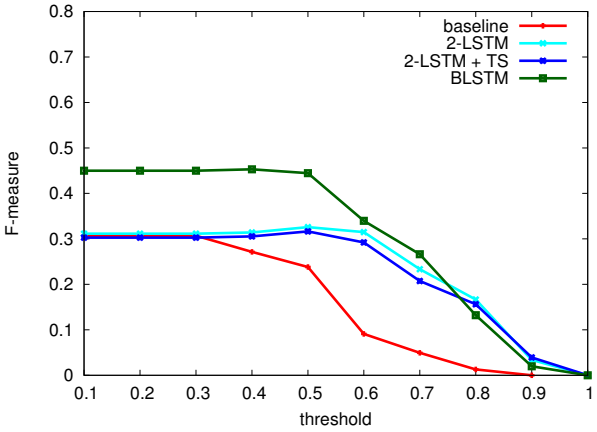

図 2: ×の検出結果

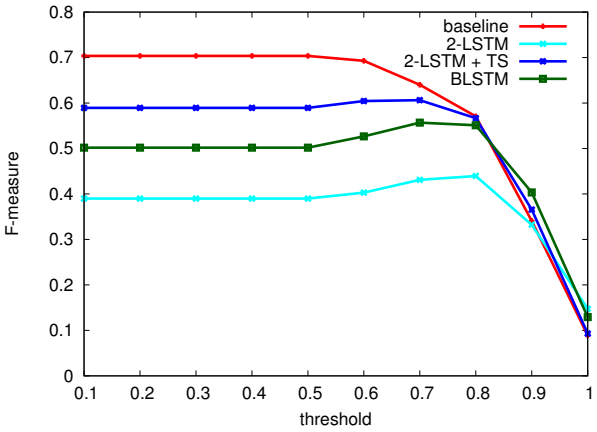

図 3: △+×の検出結果

図 2 では，$t = 0.1$ から 0.7 まで BLSTM の F 値が最も高く，$t = 0.8$ では 2-LSTM が，$t = 0.8$ では 2-LSTM + TS がそれぞれ最も高くなった．一方，ベースラインは $t = 0.4$ 以降大きく F 値が低下し，提案手法と大きく差が出る結果となった．しかし，図 3 では，3 つの提案手法が $t = 0.1$ から 0.8 までベースラインに劣る結果となった．

表 1 では，すべての項目について BLSTM が最も良い結果となった．また，2-LSTM と 2-LSTM + TS についても，ベースラインに比べ良い結果となった．これは，ベースラインは分布の推定は行わず，出力として決定したアノテーションの確率を一律で 1.0 とするため，正解の分布との差異が大きくなったためであると思われる．一方，提案手法では損失関数に Mean squared error を用い，正解の分布に近づくよう学習が行われた結果，より適切な分布の推定が可能となったと考えられる．

表 1: 分布評価

	JSD	JSD (T+X)	JSD (O+T)	MSE	MSE (T+X)	MSE (O+T)
baseline	0.403	0.258	0.202	0.215	0.226	0.164
2-LSTM	0.122	0.097	0.064	0.070	0.109	0.065
2-LSTM + TS	0.143	0.106	0.076	0.083	0.118	0.075
BLSTM	**0.118**	**0.094**	**0.058**	**0.069**	**0.108**	**0.058**

4 まとめ

本稿では，Long short-term memory を中間層に用いた Recurrent Neural Network(LSTM-RNN) による対話破綻検出手法について述べた．提案手法では，破綻検出対象となる対話システムの発話と，その直前のユーザ発話の 2 発話のみを用いた．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 LSTM-RNN の入力とした．LSTM-RNN はシステム発話用とユーザ発話用の 2 種類を用意した．システム発話から得た分散表現系列と，ユーザ発話から得た分散表現系列をそれぞれ入力し，2 つの出力を統合して対話破綻アノテーションの分布を推定した．

実験では，構成・設定の異なる 3 種類の LSTM-RNN を用いて破綻検出を行った．3 つの提案手法は×の検出とアノテーション分布の推定についてはベースラインを上回る性能を確認したが，△と×の検出ではベースラインよりも低い性能となった．

参考文献

[1] 東中竜一郎, 船越孝太郎. Project next nlp 対話タスクにおける雑談対話データの収集と対話破綻アノテーション. 言語・音声理解と対話処理研究会, Vol. 72, pp. 45–50, 2014.

[2] 東中竜一郎, 船越孝太郎, 小林優佳, 稲葉通将. 対話破綻検出チャレンジ. 第 6 回対話システムシンポジウム, 2015.

[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] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, Vol. 9, No. 8, pp. 1735–1780, 1997.

[6] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *Signal Processing, IEEE Transactions on*, Vol. 45, No. 11, pp. 2673–2681, 1997.