

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

Multi-task Learning of Recurrent Neural Network for Detecting Breakdowns of dialog and Language Modeling

小林 颯介^{1*} 海野 裕也² 福田 昌昭²
Sosuke Kobayashi¹ Yuya Unno² Masaaki Fukuda²

¹ 東北大学

¹ Tohoku University

² 株式会社 Preferred Infrastructure

² Preferred Infrastructure, Inc.

Abstract: For detecting dialog breakdowns, we make it as a classification problem using features of sentences in the dialog. The feature including context information will perform better on the task. Recurrent neural networks (RNN) can encode sentences to fixed-length feature vectors, which are based on previous contexts. In this paper, we propose a multi-task learning method of language modeling and detecting dialog breakdowns for RNN and extended models. We conducted comparative experiments for our various RNN models and learning methods on dataset of *Dialog breakdown detection challenge*, and show that our proposed model has very high precision, low recall, high accuracy and very small mean squared error for prediction of dialog breakdown labels compared to other participants.

1 はじめに

対話破綻検出チャレンジ [東中 15] では、人間とシステムが行った対話のログについて、対話が破綻している箇所を検出するタスクを行う。対話の破綻をも的確に検出できれば、対話システムが不適切な返答を避けて対話を続けることが可能になることが期待される。各発話データに対して破綻か否かのラベルが付与されているので、これを予測する問題とみなせる。

分類のための素性は、ニューラルネットワークを用いて対話文を表現ベクトルにエンコードする。そのために、文内の各単語を分散表現の単語ベクトルとして扱い、さらに構造的に文全体を固定次元の素性ベクトル変換する。ニューラルネットワークによる文や単語の分散表現へのエンコードモデルは、教師なしの対話データやコーパスから学習が可能である。対話破綻検出のためのラベル付きデータセットは規模が小さいので、教師なし学習として大量のコーパスデータを用いることができるならば、語彙や対話現象のバリエーションの狭さを回避し一般化された素性表現を得ることが

期待される。教師なしデータから学習されたニューラルネットワークで得た分散表現が、別の教師ありタスクの素性としても有効に働くことは近年確かめられてきている [Komatsu 15]。

ニューラルネットワークによる文のベクトル化については近年多くの研究がなされ [Kiros 15, Cho 14]、対話処理に関連する手法も注目を集めている [Vinyals 15, Shang 15, Lowe 15]。その中の1つであるニューラルネット対話モデル (Neural Conversational Model; NCM) [Vinyals 15] は、対話相手の発言文をまずベクトルにエンコードし、そこから返答文を生成するモデルであり、その生成内容による対話の質の高さから注目を集めている。このモデルは元は機械翻訳で高性能を示した seq2seq [Sutskever 14] を応用した手法であり、その汎用性から破綻検出にも効果的なモデルであることが期待される。そこで、本稿では、ニューラルネット対話モデルを対話破綻検出に応用するにあたって新たに拡張を行ったモデルを提案する。また、対話破綻検出タスクに対して有用な文エンコードモデルを学習するために新たな目的関数を提案し、様々なモデルについて適用した場合の性能比較実験を行った。

*連絡先： 東北大学
宮城県仙台市青葉区荒巻字青葉 6-6-05 東北大学工学研究科 電子情報システム・応物系 1 号館 6 階
sosuke.k@ecei.tohoku.ac.jp

2 関連研究

2.1 再帰型ニューラルネットワーク言語モデル

再帰型ニューラルネットワーク (Recurrent Neural Network; RNN) は、可変長のシーケンスデータから可変長のシーケンスデータへの入出力処理が可能なニューラルネットワークモデルである。さらに、その亜種として生まれた Long Short-Term Memory (LSTM)[Hochreiter 97] は、再帰型ニューラルネットワークの勾配消失問題を緩和し、性能が向上することが知られている [Graves 13]。LSTM の場合、長さ T_i の入力シーケンス $\{x_1, x_2, \dots, x_{T_i}\}$ について、長さ T_o の出力シーケンス $\{h_1, h_2, \dots, h_{T_o}\}$ を得る処理は以下のように定式化される [Graves 13]。

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \quad (1)$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \quad (2)$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \quad (3)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (4)$$

$$h_t = o_t \odot \tanh(c_t) \quad (5)$$

σ , \tanh はそれぞれシグモイド関数とハイパボリックタンジェントであり、 i_t, f_t, o_t, c_t, h_t の各ベクトルを算出するための W_* は線形写像を行う行列、 b_* はバイアス用のベクトルであり、 W_* と b_* が学習によって最適化されるパラメータとなる。

言語モデルの場合はさらに拡張が必要となる。一般には入出力ベクトルの次元数 (語彙数) と処理系内部での次元数が異なるため、以下の定式化が追加される。

$$x_t = W_{x'x}x'_t \quad (6)$$

$$y_t = \text{softmax}(W_{hy}h_t) \quad (7)$$

入力シーケンス (単語列) は $\{x'_1, x'_2, \dots, x'_{T_i}\}$ で、 x'_* は語彙数と同じ数の次元数で特定のインデックス (次元) の値のみが 1 で他が 0 となったベクトル (one-hot ベクトル) で表される。出力シーケンス $\{y_1, y_2, \dots, y_{T_o}\}$ は、各時刻における予測出力単語の確率分布を表すベクトルである。softmax はソフトマックス関数であり、出力値を確率分布へと正規化する。

言語モデルの学習では、入力単語列 $\{x'_1 (= bos), x'_2, \dots, x'_{T_i}\}$ から、それぞれの次に現れる単語を順に予測する。bos は文頭で入力される特殊なシンボル単語である。この場合正しい出力単語列は $\{y'_1 = x'_2, y'_2 = x'_3, \dots, y'_T = eos\}$ となる。eos は文末を示す特殊なシンボル単語である。正しい出力単語列の出力確率は以下ようになる。 c_0 及び h_0 は零ベクトルとする。

$$p(y'_1, \dots, y'_T | x'_1, \dots, x'_{T_i}) = \prod_{t=1}^T p(y'_t | c_{t-1}, h_{t-1}, x'_t) \quad (8)$$

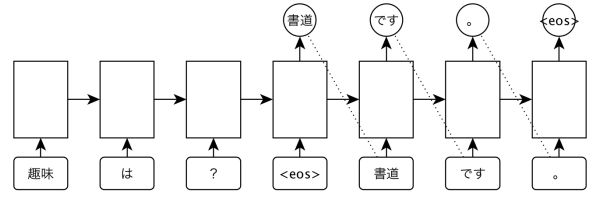


図 1: ニューラルネットワーク対話モデル

2.2 ニューラルネットワーク対話モデル

Vinyals らは、Sutskever らの seq2seq[Sutskever 14] を対話データを用い一部の拡張を行い、ニューラルネットワーク対話モデルとして提案した [Vinyals 15]。以下ではその手法の詳細を述べる。

モデルの構造は概ね 2.1 で述べた LSTM による言語モデルと同一である。そのモデル構造と入出力の例を図 1 に示す。変更点は、(a) 対話の発話人物が変わる時刻に発言終了のシンボル単語 (eos) を入力し、その時刻における出力で次の「返答文」の一単語目を予測する点、及び (b) 学習の際に予測する出力単語列を「返答文」に限定する点である。すなわち、先の書き方に従うと、相手の発言文の単語列を $\{x'_{i1}, \dots, x'_{iT_i}\}$ 、次の返答文の単語列を $\{x'_{o1}, \dots, x'_{oT_o}\}$ とした場合、入力単語列 $\{x'_{i1}, \dots, x'_{iT_i}, eos, x'_{o1}, \dots, x'_{oT_o-1}, x'_{oT_o}\}$ から、 $\{\epsilon, \dots, \epsilon, x'_{o1}, x'_{o2}, \dots, x'_{oT_o}, eos\}$ を予測するような学習を行うモデルだと言える。 ϵ は、予測を行う必要がなく学習の際に予測誤差を計算しない単語を表している。

3 提案手法

3.1 対話破綻検出

対話破綻検出タスクのデータは、人間が対話システムと行った対話のログデータが元になっている。人間の発話 (相手文) とシステムの発話 (返答文) が交互に出現し、各「返答文」には破綻しているか否かのラベルが複数付与されている。このデータを用いて、対話破綻検出タスクを次のように定式化する。

入力 「相手文」と「返答文」の文ペア

出力 「返答文」についての 3 種の破綻ラベル確率

すなわち、システムの返答文と直前の文のみを入力とし、それ以前の文脈は一切無視する。データセットの中には 3 文以上の広い文脈を考慮する必要がある問題も存在しうるが、比較的簡単とみられる狭い文脈内での問題のみに焦点を当てて取り組むこととする。

そして、対話破綻検出のタスクを以下の 2 つの処理に分解する。

1. 文エンコード「相手文」と「返答文」をそれぞれ固定長次元のベクトル v_i, v_o に変換する
2. ラベル推定 v_i と v_o を用いて 3 種の破綻ラベル確率を推定する

2. には, v_i と v_o と両者のベクトルの要素積 $v_i \odot v_o$ を結合して作成される素性ベクトルを用いて, 以下の様な 2 層のニューラルネットワークを用いる。

$$d_1 = \sigma(W_{d1}[v_i; v_o; v_i \odot v_o]) \quad (9)$$

$$p_d = \text{softmax}(W_{d2}d_1) \quad (10)$$

$[v_i; v_o; v_i \odot v_o]$ は, 3 つのベクトルの結合を表す。学習時には上記のニューラルネットワークを用いて, 文ベクトルペア (v_i, v_o) と正解ラベル確率 p_d の組からなる教師ありデータ集合 S に対して, 以下の目的関数を最小化する。

$$-\sum_{(v_i, v_o, p_d) \in S} \sum_{l=O, T, X} p_d^l \log \hat{p}_d^l \quad (11)$$

p_d^l, \hat{p}_d^l は, それぞれラベル l に対する正解の確率と予測上の確率であり, すなわち上の目的関数は正解ラベル確率と予測ラベル確率の交差エントロピー誤差になっている。なお, モデル全体としては 1 の学習後にそれらのパラメータを固定した上で追加で 2 の学習を行った。

本稿の主題は, 1. の文エンコードの手法にある。ここでは複数のモデルが考えられる。また, 文エンコードモデルの学習での目的関数についても複数考えられるため, それらを組み合わせ的に比較する。以下でモデルと学習法についてそれぞれ詳細を説明する。

文ペアに対して文ベクトルペアを生成する文エンコードモデルとして, 以下の 4 種類のモデルを用意した。後の実験結果の表 1 内での略称も括弧書きで載せる。また, 各モデルの比較図を図 2 に示す。

- A. NCM NCM による文確率算出の手順をなぞり, 相手文を入力後の出力ベクトル h_{T_i} を相手文ベクトル, 返答文を入力後の出力ベクトル h_{T_o} を返答文ベクトルとする [NCM]
- B. LSTM エンコーダ 相手文, 返答文それぞれを単に LSTM でエンコードする (NCM で相手文から返答文への連結を無視したモデル) [個別]
- C. Bag-of-Words Embedding 相手文, 返答文それぞれについて, 含まれる単語の Word Embedding の平均をとることで文ベクトルとする [BOW]
- D. NCM の拡張モデル 文ベクトルの決定は NCM と同様。モデルの詳細は 3.2 で説明を行う [提案]

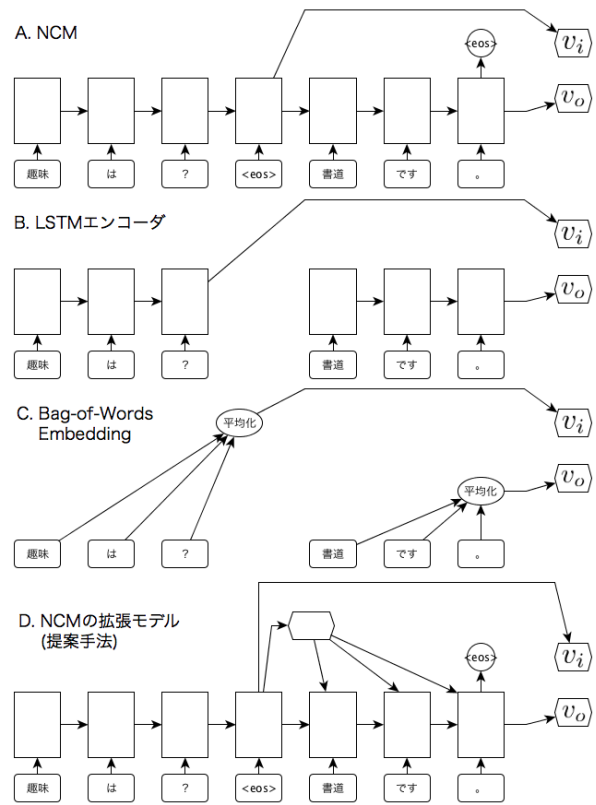


図 2: 文エンコードモデルの比較

D. のモデルは本稿での新たな提案モデルである。3.2 にてその説明を行う。

文エンコードモデルの学習の目的関数として以下の 3 種類を用意した。

- a. 単語予測学習 2.1., 2.2. で述べた学習。C を除くモデルで可能。
- b. 文ベクトルの内積最大化 相手文, 返答文ベクトルのペアから算出する非破綻確率を最大化するような学習を行う。具体的には, 相手文, 返答文ベクトルをそれぞれ単層の NN で変換後, それらの内積をシグモイド関数に通したものを非破綻ペア (正例) について最大化, 破綻ペア (負例) について最大化する。負例については, 対話コーパス中出现する相手文と返答文のペアについて, 返答文をランダムに入れ替えたものを正例 1 つに対して 5 つ作成して用いた。そのような負例コーパスを \bar{S} とおくと, 最小化を行う損失関数は以下の式 12 を用いた。

$$-\left(\sum_{(v_i, v_o) \in S} \log \sigma(v_i' v_o') + \sum_{(v_i, v_o) \in \bar{S}} \log \sigma(-v_i' v_o') \right) \quad (12)$$

$$v_*' = \tanh(W_*' v_*), \quad * = i, o \quad (13)$$

- c. マルチタスク学習 上の a., b. の両方について並列的に学習を行う。最小化する目的関数は両者の目的関数の（重み付けのない）和とした

3.2 ニューラルネット対話モデルの拡張

seq2seq を始めとして, LSTM による文エンコードモデルは機械翻訳 [Cho 14] を中心に研究が盛んである。また, 入力となる文のベクトルをエンコードした後に別の文を生成する際に, 毎時刻で 1 文目のベクトルを注入し直すという処理を行うことで, 1 文目の文脈情報を覚えるための LSTM 内の隠れ層ユニットの記憶効率が上がり性能が向上する可能性が示唆されている。このモデルの一部を利用して NCM を新たに拡張する。

以上のような考えにもとづき, NCM を拡張した新たな提案モデルを以下のように定義する。

まず, 相手文は (1) ~ (6) の式によって定義される LSTM エンコードモデルによって一単語ずつ入力を行っていき, 文末単語の次に文末シンボルである *eos* を入力した時刻での出力 h_{T_i} を相手文ベクトルとする。そして, 相手文ベクトル h_{T_i} と対応する返答文の単語列 $\{x'_1, \dots, x'_{T_o}\}$ について以下の式に基いて処理を行い, 最終単語 x'_{T_o} を入力した際の出力 h_{T_o} を返答文ベクトルとする。

$$i_t = \sigma(W'_{xi}x_t + W'_{hi}h_{t-1} + W'_{ui}h_{T_i} + b'_i) \quad (14)$$

$$f_t = \sigma(W'_{xf}x_t + W'_{hf}h_{t-1} + W'_{uf}h_{T_i} + b'_f) \quad (15)$$

$$o_t = \sigma(W'_{xo}x_t + W'_{ho}h_{t-1} + W'_{uo}h_{T_i} + b'_o) \quad (16)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh(W'_{xc}x_t + W'_{hc}h_{t-1} + W'_{uc}h_{T_i} + b'_c) \quad (17)$$

$$h_t = o_t \odot \tanh(c_t) \quad (18)$$

4 実験

対話破綻検出モデルの性能を検証するために, 2 つの比較実験を行った。1 つ目は, (a) ニューラルネットワークを用いたモデル内での比較実験 (3. で述べたモデルと学習の組み合わせについての性能の検証), 2 つ目は (b) 対話破綻検出チャレンジの公式なテストデータ提出である。これにより, 提案モデルが対話破綻検出チャレンジでの他参加者のモデルと比べてどのような特徴を持つのかを分析した。

4.1 実験設定

4.1.1 データセット

文エンコードモデルの学習コーパスとして, Web 上の匿名掲示板に投稿された二次創作短編小説から抽出

した対話データを用いた。元のデータから, 二人のみでの対話と見なせるもののみを抽出し, さらに語彙を頻出単語 2 万語に制限し, 人物名を既定のシンボル単語に置換し, 最終的に相手文と返答文のペアが合計約 30 万対となった学習データを作成した。

対話破綻検出チャレンジの教師ラベルデータセット¹ は, ラベル付きの返答文の数としては, 訓練データが 9620 個, 開発用データが 220 個, 提出用のテストデータが 880 個だった。今回は, (a) の比較実験では, 訓練データで学習を行い, 開発用データで性能評価を行った。ただし, 訓練データでの学習は, 最大 20 周まで 2 周ごとに開発用データでの損失関数を求め, 初めて損失関数が上昇した時点で学習を打ち切り, 開発用データをそのまま用いて性能を評価した。一方で, (b) の提出テスト用の学習では, (a) と同様の学習を行い, テストデータを用いて性能を評価した。

4.1.2 学習設定

単語の埋め込みベクトルの次元数と LSTM の内部の各隠れ層及び出力層のユニット数は 1000 に固定した。また, ラベル推定モデルのネットワークでは, 3000 次元-500 次元-3 次元 (3 種のラベル確率) という構成にした。文エンコードモデル学習時には学習コーパス全体を 10 周させた²。全パラメータは $[-\frac{6}{\sqrt{1000}}, \frac{6}{\sqrt{1000}}]$ の一様分布に基づいて初期化した。最適化アルゴリズムには Adam [Kingma 15] を用いた。ミニバッチのサイズは 128 にし, 学習中にはミニバッチごとの勾配を求めたのちにバッチサイズ (128) でその勾配を除算し, さらに最大ノルムを 5 とした勾配のクリッピングを行った。内積最大化の学習時には, 正例 1 つごとに 5 つの負例を作成したが, 学習処理の高速化のために今回はサンプリングを各ミニバッチ内に限定して行った。さらに学習処理の高速化のために, 各ミニバッチは「相手文」の単語数が同数であるような対話文ペアのみがまとめてミニバッチに選ばれるようにした。なお, 本モデルの実装・学習は Chainer³ を用いて行った。

4.2 実験結果

対話破綻データセットのサイズが小さいため, 各モデルで文エンコードモデルを学習後に, ラベル推定モデルの学習と評価を 10 回繰り返し, その 10 回の結果のマクロ平均を各モデルの性能として記録した。なお, 対話破綻ラベルには O, T, X の 3 種があるが, T と

¹<https://sites.google.com/site/dialoguebreakdown-detection/chat-dialogue-corpus>

²事前の予備実験により NCM を今回のコーパスで学習させた場合に, 訓練データとは異なる検証用データに対して単語予測誤差が 10 周付近で最小になることを確認した。

³<http://chainer.org/>

表 1: モデルの比較実験結果

学習法	マルチタスク学習			単語予測のみ			内積最大化のみ			
	提案@	NCM	個別	提案	NCM	個別	提案	NCM	個別	BOW
Accuracy	.495	.489	.480	.502	.502	.497	.462	.449	.471	.453
Precision (X)	.706	.556	.603	.657	.681	.507	.307	.259	.447	.650
Recall (X)	.121	.115	.111	.146	.129	.111	.019	.026	.056	.032
F-measure (X)	.204	.185	.182	.234	.209	.174	.035	.047	.098	.060
Precision (T+X)	.885	.914	.918	.907	.920	.936	.890	.823	.861	.879
Recall (T+X)	.206	.249	.239	.279	.272	.288	.044	.161	.127	.084
F-measure (T+X)	.333	.390	.378	.426	.412	.439	.083	.266	.220	.152

X の判別は人でも判断が別れるため、それらを同一のラベルとしてみなした場合についても評価を行った。その比較実験の結果を表 1 に示す。ラベル X に対する Precision はマルチタスク学習が、一方で Recall と F 値は単語予測のみを行った場合が最も高かった。ラベル T と X を同一にみなした場合では、2 文それぞれに対して個別にエンコードを行う方法が特に性能が良く、マルチタスク学習ではなく単語予測学習のみを行った場合に Precision, Recall, F 値それぞれで最高性能となった。全体として、文エンコードモデルについては、X 単一ラベルについての破綻検出性能は提案モデルが高く、T+X については個別エンコードモデルが高かった。また、学習法については、単語予測のみを行う場合でも性能が高い傾向が見られた一方で、内積最大化のみを行う学習では性能が大きく低下した。

全体としてどのモデル・学習でも共通して高 Precision, 低 Recall の傾向が見られた。これはそもそもの学習データセットのラベルにおいて、X や T の比率が O に比べて少なめであることにも起因すると考えられるが、それを差し引いても本モデルの特徴的な点だと考えられる。したがって、提出用のテストでは、その特徴が色濃く結果に現れた、マルチタスク学習を行った提案モデル (表 1 中の@) を用いた。

テストデータセットにおける実験結果を、他チーム (run) の結果と合わせて表 2 に示す。この結果は、アノテーションでのラベル一致率が 50% 以上のデータのみに対する結果である。提案モデルは、他チームと比較した場合についても非常に高い Precision と低い Recall が特徴的だと言える。破綻ラベルを X のみとした場合でも T+X とした場合でも共通して最高の Precision を達成した。また、提案モデルは、多数派ラベルである O を多く予測する傾向にあるモデルとはいえ、Accuracy でも 15 個の run の中で 4 位となった。さらに特筆すべき点として、ラベル確率の予測誤差の低さが挙げられる。表 2 に示した平均二乗誤差 (MSE) では、3 つ中 2 つのラベル分割パターンにおいて最小の誤差となった。表には載せなかったが、JS divergence についても同じ順位となった。

5 考察

今回、前後の文脈を用いず文ペアのみを入力としたため、文ペアとしては対話が破綻していないが、より広い文脈を見た場合には破綻しているような対話に対して非破綻ラベルの O を予測してしまう可能性が高い。今回の実験結果から本モデルが非常に高い Precision と低い Recall を持つモデルだと判断できるが、今回の簡略化された問題設定自体もその原因のひとつとして考えられる。

文エンコードモデルとして、対話返答文の生成モデルである NCM などを転用した。しかし単語予測誤差最小化によって学習する場合、最後の出力隠れ層は理想的には「文終了シンボル」の確率最大化を目指すようなベクトルになっているにすぎず、隠れ層に含まれる情報が返答文全体の性質として適当なものであるとは考え難い。返答文ベクトルとしては、文生成時の毎時刻の隠れ層を利用するなどの拡張が必要であろう。

NCM の拡張モデルを提案するだけでなく、それらに対して単語予測誤差最小化以外に対話破綻検出用の目的関数も設計したが、性能の向上は見られなかった。今回の実験では、ハイパーパラメータの探索や、負例の作成方法についても十分な検証を行えなかったため、今後改善する必要があるだろう。

最後に、今回の文エンコードモデルでは全てランダムな初期値から学習を始めた。この初期値のうち単語入力用のベクトルを GloVe [Pennington 14] などの大規模コーパスで学習済みのベクトルで置き換えることで性能の向上が期待される。

6 まとめ

本稿では、対話破綻検出チャレンジのデータセットを用いて、ニューラルネットワーク対話モデルを拡張したモデルを始めとした複数のモデルの比較評価を行った。また、他の様々な手法に比べて極めて高い Precision や小さいラベル予測誤差を達成することを示した。今後は破綻検出モデルとしての最適化学習についてより詳細な分析改善を行う必要がある。

表 2: 他チームのモデルとの比較実験結果

Team-run	1-1	1-2	2-1	2-2	2-3	3-1	3-2	3-3	4-1	4-2	4-3	5-1	5-2	5-3	提案	順位
O(個数)	562	873	642	561	715	498	498	767	733	810	830	419	283	268	799	4
T(個数)	11	0	40	220	50	340	0	0	0	0	0	163	132	314	49	7
X(個数)	307	7	198	99	115	42	382	113	147	70	50	298	465	298	32	14
Acc	.530	.628	.643	.574	.627	.460	.515	.570	.599	.633	.627	.515	.477	.442	.631	3
P(X)	.277	.429	.444	.475	.443	.167	.288	.221	.313	.414	.380	.366	.333	.366	.531	1
R(X)	.429	.015	.444	.237	.258	.035	.556	.126	.232	.146	.096	.551	.783	.551	.086	13
F(X)	.337	.029	.444	.316	.326	.058	.379	.161	.267	.216	.153	.440	.468	.440	.148	13
P(T+X)	.758	.857	.824	.796	.836	.725	.725	.628	.741	.800	.820	.790	.760	.753	.926	1
R(T+X)	.444	.011	.361	.468	.254	.510	.510	.131	.201	.103	.076	.670	.836	.849	.138	11
F(T+X)	.560	.022	.502	.589	.390	.599	.599	.216	.316	.183	.138	.725	.796	.798	.240	11
MSE (O,T,X)	.104	.200	.069	.083	.070	.211	.229	.239	.224	.218	.220	.203	.204	.208	.060	1
MSE (O,T+X)	.133	.366	.108	.118	.109	.278	.280	.401	.362	.383	.393	.213	.172	.173	.090	1
MSE (O+T,X)	.105	.123	.058	.075	.065	.143	.289	.197	.191	.152	.148	.213	.271	.213	.062	2

謝辞

対話破綻検出のデータセット作成に携わった皆さま、創作小説の匿名投稿者の皆さま、ならびに論文執筆にあたって有益な議論をして頂いた株式会社 Preferred Infrastructure, Preferred Networks 及び東北大学の乾・岡崎研究室の皆さまに感謝致します。

参考文献

- [Cho 14] Cho, K., Merriënboer, van B., Gülçehre, Ç., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y.: Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation, in *Proc. of EMNLP 2014*, pp. 1724–1734 (2014)
- [Graves 13] Graves, A.: Generating Sequences With Recurrent Neural Networks, *arXiv preprint arXiv:1308.0850* (2013)
- [Hochreiter 97] Hochreiter, S. and Schmidhuber, J.: Long Short-Term Memory, *Neural Computation*, Vol. 9, No. 8, pp. 1735–1780 (1997)
- [Kingma 15] Kingma, D. P. and Ba, J.: Adam: A Method for Stochastic Optimization, in *Proc. of ICLR 2015* (2015)
- [Kiros 15] Kiros, R., Zhu, Y., Salakhutdinov, R., Zemel, R. S., Torralba, A., Urtasun, R., and Fidler, S.: Skip-Thought Vectors, *arXiv preprint arXiv:1506.06726* (2015)

[Komatsu 15] Komatsu, H., Tian, R., Okazaki, N., and Inui, K.: Reducing Lexical features in Parsing by Word Embeddings, *Proc. of PACLIC 2015* (2015)

[Lowe 15] Lowe, R., Pow, N., Serban, I., and Pineau, J.: The Ubuntu Dialogue Corpus: A Large Dataset for Research in Unstructured Multi-Turn Dialogue Systems, in *Proc. of SIGDIAL 2015*, pp. 285–294, Prague, Czech Republic (2015), Association for Computational Linguistics

[Pennington 14] Pennington, J., Socher, R., and Manning, C. D.: Glove: Global Vectors for Word Representation, in *Proc. of EMNLP 2014*, pp. 1532–1543 (2014)

[Shang 15] Shang, L., Lu, Z., and Li, H.: Neural Responding Machine for Short-Text Conversation, in *Proc. of ACL 2015*, pp. 1577–1586 (2015)

[Sutskever 14] Sutskever, I., Vinyals, O., and Le, Q. V.: Sequence to Sequence Learning with Neural Networks, in *Proc. of NIPS 2014*, pp. 3104–3112 (2014)

[Vinyals 15] Vinyals, O. and Le, Q. V.: A Neural Conversational Model, in *Proc. of ICML 2015 Deep Learning Workshop* (2015)

[東中 15] 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)