人工知能学会研究会資料
SIG-SLUD-B502-12

異なる特性を持つデータの組み合わせによる雑談対話の破綻検出

Chat-oriented Dialogue Breakdown Detection based on Combination of Various Data

杉山 弘晃 [1]
* Hiroaki Sugiyama [1]

[1] NTT コミュニケーション科学基礎研究所
[1] NTT Communication Science Laboratories

Abstract:
Chat-oriented dialogue systems sometimes generate utterances that are inappropriate as the responses for user utterances and cause dialogue breakdown If a system can predict whether an utterance cause dialogue breakdown, it helps to continue dialogue with suppressing such inappropriate system utterances. In this paper, I develop a dialogue breakingdown detector and analyze the effects of training features, data and algorithms for dialogue breakdown detection performance.

1 序論

近年，従来のタスク指向の対話システムとは異なる，雑談を行う対話システムに注目が集まっている [大西 14, Ritter 11, Wong 12, 東中 14a]．雑談対話は，エンタテインメントやカウンセリング目的のみならず，ユーザの潜在的な要求を引き出したり，ユーザと良好な関係を構築する上で重要である．

雑談対話システムは，ユーザ発話に含まれる非常に幅広い話題に応答する必要がある．そのため，適切な応答を出力し続けることは難しく，現在の対話システムでは，対話を破綻させるような発話がしばしば生成される．こうした破綻する可能性のある発話を予め検出し，出力を抑制することができれば，対話の継続が容易になると考えられる．

こうした背景から，東中らは，人とシステムの雑談対話に含まれる破綻箇所を人手で付与した，雑談対話破綻コーパス [東中 14b] を公開している．雑談対話破綻コーパスを用いることで，雑談対話の破綻検出器を容易に評価できるようになる．東中らは，この雑談対話破綻コーパスを利用し，雑談対話の破綻検出器の性能を競う，対話破綻検出チャレンジを開催している [東中 15]．

本研究では，筆者らがこれまで収集してきた多様なデータを用いて破綻検出器を構築し比較することで，雑談対話の破綻検出器を構築する上で有用な特徴量，訓練データ構成，アルゴリズムについて分析する．

*連絡先：NTT コミュニケーション科学基礎研究所
〒 619-0237 京都府相楽郡精華町光台 2-4
E-mail: sugiyama.hiroaki@lab.ntt.co.jp

2 比較要素

本節では，比較に用いた特徴量，訓練データ，アルゴリズムの詳細を説明する．

2.1 特徴量

本研究では，識別に有用な特徴を明らかにするため，以下の全特徴量を用いた検出器と，いずれかの特徴量を用いずに構築した検出器との，破綻検出性能を比較する．

単語　破綻検出の対象となる，ある対話中の t 番目のシステム発話文 s_t，およびその直前のユーザ発話文 s_{t-1} に含まれる単語 unigram の Bag-of-words をつなげたベクトル．単語辞書は，タスク主催者から配布された rest1046（2.3 節に後述）に含まれる 7186 単語で構築し，2 発話計 14372 次元のベクトルを構築する．なお，単語の出現数による辞書のフィルタリングは行わない．

単語クラス　破綻検出の対象となるシステム発話文 s_t，およびその直前のユーザ発話文 s_{t-1} に含まれる単語ク

ラスの Bag-of-words をつなげたベクトル．単語クラスは word2vec ベクトルを k-means でクラスタリングしたもの．k-means の k は実験的に 2000 とし，2 発話分 4000 次元のベクトルを得る．

単語組み合わせ　破綻検出の対象となるシステム発話文 s_t，およびその直前のユーザ発話文 s_{t-1} に含まれる単語 $w_{t,i} \in W_t, w_{t-1,j} \in W_{t-1}$ について，取りうる組み合わせ $\{w_{t,i}, w_{t-1,j}\}$ を列挙したうち，学習データ中の出現数が上位 N 個の Bag-of-words ベクトル．本研究では，実験的に $N = 500$ とした．

単語クラス組み合わせ　破綻検出の対象となるシステム発話文 s_t，およびその直前のユーザ発話文 s_{t-1} に含まれる単語の単語クラス $c_{t,i} \in C_t, c_{t-1,j} \in C_{t-1}$ について，取りうる組み合わせ $\{c_{t,i}, c_{t-1,j}\}$ を列挙したうち，学習データ中で頻出した（出現数上位 N 個の）組み合わせが含まれているか否かを表す 1 次元のフラグ．単純に組み合わせの Bag-of-words ベクトルを利用するよりも，予備実験で良い性能を示していたため，フラグでの表現を用いる．本研究では，実験的に $N = 500$ とした．

発話自動評価値　杉山らが提案した，雑談対話システムの自動評価に用いられる，大規模マルチリファレンスコーパス [杉山 14a] を用いて推定された，発話の評価値．この大規模マルチリファレンスコーパスは，Twitter から 200 文，人同士の雑談対話コーパス [Higashinaka 14] から 300 文，計 500 文をランダムに選んで入力文とし，その各々に対して作成された 100 文の応答文について，6 人の評価者が 7 段階の主観評価値（応答文としての自然さ）を付与したものである．本研究では，このコーパスを用いて，ある入力文に対する応答文の評価値を予測するモデルを，線形カーネルの SVR を用いて構築し，直前のユーザ発話文 s_{t-1} を入力文，破綻検出の対象となるシステム発話文 s_t を応答文として得られた推定評価値を特徴量に用いる．本推定器が出力した評価値は，人が付与した主観評価値に対し，0.450 の相関係数を示していた．

パープレキシティ　破綻検出の対象となるシステム発話文 s_t，およびその直前のユーザ発話文 s_{t-1} に対し，Keyser-ney スムージングをかけて単語 4gram で計算したパープレキシティから，1gram の値を引いたもの．パープレキシティは文の流暢さを表現する目的で導入するが，単語 4gram のみでは，パープレキシティが低い場合に，単に出現数の少ない単語が多いのか，単語間の接続確率が低いのかがわからない．そのため，1gram での値を引くことで単語自体の出現確率を正規化し，単語間の接続確率に焦点を当てた値としてを利用する．計算にはブログデータ（2009 年 1 月から 2012 年 2 月にクロール）を用い，言語モデルの計算は kenlm[1] で行った．

直前発話との類似度　破綻検出の対象となるシステム発話 s_t と，その直前のユーザ・システム発話 s_{t-1}, s_{t-2} との単語コサイン類似度．対話破綻コーパス内の発話を分析すると，システムがシステム自身やユーザの直前の発話とほぼ同じ意味の文を繰り返し発話し，破綻と評価されている例が多く見られた．この類似度特徴は，そうした繰り返し発話の検出に有用であると考えられる．

パーソナリティ質問　破綻検出の対象となるシステム発話文 s_t，およびその直前のユーザ発話文 s_{t-1} が，システム自身の具体的な嗜好や経験を問う，パーソナリティ質問か否かを推定したフラグ [杉山 14b]．対話破綻コーパス内には，ユーザがシステム自身のことを尋ねていてもシステムが答えられず，破綻と評価されている例が見られたており，本特徴量でそうした例を検出できると考えられる．推定器には単語を特徴量とする LinearSVM を用い，杉山らが構築した Person DataBase[Sugiyama 14]，および上述の雑談対話コーパスに付与された，パーソナリティ質問か否かを人手で分類したフラグを利用して学習する．パーソナリティ質問か否かの推定精度は，マジョリティベースラインで 0.770 のところ，0.988 であった．

対話行為　破綻検出の対象となるシステム発話文 s_t の対話行為の確率を並べたベクトル $p(a_{t,i}), a_{t,i} \in A_t$，その直前のユーザ発話文 s_{t-1} の対話行為の確率を並べたベクトル $p(a_{t,i}), a_{t-1,i} \in A_{t-1}$，および直前のユーザ発話文から予測されたシステムが出すべき対話行為の確率を並べたベクトル $\hat{a}_{t,i}, \hat{a}_{t,i} \in \hat{A}_t$．対話行為とは，質問や自己開示など，発話が意図する行為を大まかに分類したものである．本研究では，目黒ら [Meguro 10] で定義された，33 種類の対話行為から成るセットを用いる．対話行為の推定器には，単語特徴を利用した線形カーネルの SVM を用いる．推定器の学習は，人同士

[1] https://kheafield.com/code/kenlm/

の雑談対話コーパス [Higashinaka 14] の各発話に人手で付与された対話行為を利用する．

2.2 アルゴリズム

本研究では，多数の疎な特徴量を扱えるアルゴリズムとして，線形カーネル SVM と DNN を比較する．線形カーネル SVM のパラメータは実験的に決定する．DNN は多層パーセプトロンで構成し，活性化関数には ReLU，出力レイヤに softmax を用いる．DNN の学習は，50%のニューロンを dropout させ，pre-train をせずに，AdaGrad で最適化する．誤差関数には平均自乗誤差を用いるが，訓練ラベルは評価の割合ではなく，$\{1, 0, 0\}$ のようなバイナリ値で付与する．本研究では，表 1 のように層の構成を変化させて破綻検出性能を比較する．なお，SVM の実装は scikit-learn[2] を，DNN の実装は chainer[3] を利用する．

表 1: 比較する DNN の構造．層構造は，入力層と出力層の値を省いたパーセプトロンの各層のニューロン数を表す

Name	層構成
DNN1	100
DNN2	100-100
DNN3	1000-1000
DNN4	5000-5000
DNN5	5000-5000-5000
DNN6	5000-5000-5000-5000
DNN7	7500-5000-2500-2500

2.3 訓練データ

検出器の訓練データには，rest1046 を基本として，以下のデータを利用する．

rest1046 破綻検出チャレンジで配布された，学習用の 1046 対話（システム発話文は 11506 文）．各発話について 2 名の評価者が○△×を付与した．この 2 名は○を付与しやすい評価者と，×を付与しやすい評価者の組み合わせとなるよう設定されている [東中 15]．2 名が異なる評価を付与した場合，ベースラインプログラムでは○を優先している．しかし，予備実験では，○を優先すると×のデータが極端に少なくなり， Recall 低下の原因となっていた．そのため，本研究では×を優先して学習時の正解ラベルとし，Recall と f 値の低下を防ぐ．

雑談システム自動評価用の大規模マルチリファレンスコーパス 上述の雑談対話システムの自動評価用に構築された大規模マルチリファレンスコーパス [杉山 14a] に含まれる，応答文とその主観評価値を利用したデータ．あるリファレンスの 6 人の評価者の平均値 x が $5 \leq x \leq 7$ ならば○，$3 \leq x \leq 5$ ならば△，$1 \leq x < 3$ ならば×とする．本コーパスは一問一答形式となっているため，訓練データへ追加する際は，各入力・応答対を 1 対話として追加する．500 入力文 ×100 応答文の計 50000 文が訓練データへ追加される．

Person DataBase 杉山らが収集している，システム自身の具体的な嗜好や経験を問う，パーソナリティ質問とその回答を収集したデータベース [杉山 14b, Sugiyama 14] に含まれる，26595 個の質問文・応答文のペアが訓練データへ追加される．マルチリファレンスコーパス同様，本 DB も一問一答形式となっているため，各質問・応答対を 1 対話として追加する．

3 実験

前章で説明した特徴量，データ，アルゴリズムを組み合わせ，破綻検出器を構成し，破綻検出性能を比較する．実際に破綻検出器を対話システムへ適用することを考えると，できるだけ破綻になりそうな発話を取りこぼさないことが重要である．一方で，全てを破綻としてしまうと，システムが発話できないことになってしまう．そのため本研究では，評価尺度はテストデータにおける f 値（X）と f 値（X+T) を用いる．ここで，f 値（X）は×のみについて， f 値（X+T）は×と△を同一視した際の f 値を表す．なお本研究では，モデル学習の目的関数を計算する際には 2.2 で説明した訓練データのみを用い，開発データにおける f 値（X or X+T）の値は，SVM ではパラメータ C の値，DNN では学習を打ち切るエポック数の決定に用いる．DNN は局所最適解に陥りやすく，かつ初期値の影響を大きく受けるため，5 回試行して開発データ上で最適なモデルを選び，そのモデルのテストデータでの値を評価に用いる．また，各 f 値の計算には，対話破綻検出チャレンジで

[2] http://scikit-learn.org/
[3] http://chainer.org/

表 2: ベースラインの性能

学習器	acc	pr(X)	rc(X)	f(X)	pr(XT)	rc(XT)	f(XT)	mse
配布（CRF ベース）	0.450	**0.625**	0.158	0.253	0.760	0.622	0.684	0.230
単語 SVM	**0.554**	0.495	**0.482**	**0.488**	**0.832**	0.605	0.701	**0.189**
全て X	0.285	0.285	1.000	0.443	0.617	**1.000**	**0.763**	0.327

表 3: 学習器の性能比較（モデル選択：f 値（X)，特徴量：全特徴量）

学習器	acc	pr(X)	rc(X)	f(X)	pr(XT)	rc(XT)	f(XT)	mse
SVM	**0.559**	0.496	0.549	0.521	0.814	0.639	0.716	**0.186**
DNN1	0.548	**0.521**	0.454	0.485	0.844	0.429	0.569	0.191
DNN2	0.547	0.492	0.494	0.493	**0.849**	0.394	0.538	0193
DNN3	0.548	0.451	0.661	0.536	0.812	0.630	0.710	0.190
DNN4	0.543	0.450	0.677	0.541	0.806	0.621	0.701	0.190
DNN5	0.553	0.467	**0.681**	**0.554**	0.805	0.646	0.717	0.189
DNN6	0.531	0.443	0.629	0.520	0.804	0.663	0.727	0.195
DNN7	0.518	0.436	0.629	0.515	0.777	**0.751**	**0.764**	0.201

表 4: 学習器の性能比較（モデル選択：f 値（X+T)，特徴量：全特徴量）

学習器	acc	pr(X)	rc(X)	f(X)	pr(XT)	rc(XT)	f(XT)	mse
SVM	0**.559**	0.496	0.549	0.521	0.814	0.639	0.716	**0.186**
DNN1	0.549	**0.531**	0.446	0.485	0.817	0.525	0.639	0.192
DNN2	0.559	0.478	0.606	0.534	**0.827**	0.484	0.611	0.191
DNN3	0.541	0.460	0.753	**0.571**	0.782	0.735	0.758	0.196
DNN4	0.458	0.408	0.693	0.514	0.725	0.893	0.800	0.223
DNN5	0.519	0.440	0.765	0.559	0.763	0.869	**0.812**	0.199
DNN6	0.450	0.348	**0.952**	0.510	0.722	**0.913**	0.807	0.234
DNN7	0.474	0.358	0.924	0.516	0.745	0.890	0.811	0.223

配布された開発・テストデータおよび評価スクリプト [東中 15] を利用する．

本研究では，比較のベースラインとして，破綻検出チャレンジの主催者から配布された CRF ベースの検出器，検出対象のシステム発話文とその直前の発話文の単語のみを利用した SVM による検出器，および全てを× とした場合の値を用いる．表 2 に，各ベースラインの結果を示す．表には，主催者から配布された評価システムが出力する，Accuracy（acc），Precision（pr），Recall（rc），f-value（f），平均自乗誤差（mse）を示している．f 値（X) では単語 SVM が，f 値（X+T）では全て× とした場合の値が高い．特に，後者の f 値（X+T）は，実用上意味を成さないベースラインに関わらず，他のベースラインとの差が大きく，この値を上回ることが実用を見据えた上での一つの目標となる．

3.1 アルゴリズムの比較

線形カーネル SVM と DNN の破綻検出性能を比較する．DNN は，表 1 に示す，層の数とニューロン数が異なる複数の構造について検証する．また，訓練データには rest1046 のみを用い，特徴量は上述の全特徴量を用いた場合を対象とする．

結果を表 3,4 に示す．SVM は f 値（X）についてはベースラインを上回っていたが，f 値（X+T）については下回っている．DNN とベースライン，SVM の比較では，層・ニューロン数が少ない DNN（DNN1,2,3）は，ベースラインや SVM を下回る場合が多い．しかし，層･ニューロン数がを増やした DNN（DNN4,5,6,7）は，いずれの f 値においても，ベースラインを大きく上回っており，特にモデルが複雑になるほど Recall の改善が大きい．このことから，DNN が持つ，複数の特徴量を組み合わせて認識を行う性質が，検出性能，特に Recall の向上に寄与したと考えられる．DNN 内で

表 5: 特徴量の追加による性能の変化（DNN7）

特徴量	acc	pr(X)	rc(X)	f(X)	pr(XT)	rc(XT)	f(XT)	mse
単語	0.478	0.363	0.944	0.524	0.747	0.899	0.816	0.224
全特徴量	0.474	0.358	0.924	0.516	0.745	0.890	0.811	0.223
-単語	0.484	0.366	0.837	0.510	0.749	0.790	0.769	0.220
-単語クラス	0.477	0.384	0.845	0.528	0.745	0.908	0.818	0.219
-単語組み合わせ	0.451	0.347	0.964	0.510	0.726	0.934	0.817	0.234
-自動評価値	0.482	0.381	0.876	0.531	0.747	0.893	0.814	0.217
-パープレキシティ	0.428	0.336	0.960	0.497	0.712	0.941	0.810	0.244
-直前発話との類似度	0.450	0.346	0.928	0.504	0.725	0.899	0.803	0.234
-パーソナリティ質問フラグ	0.495	0.387	0.896	0.541	0.759	0.902	0.824	0.211
-対話行為	0.427	0.337	0.984	0.502	0.711	0.961	0.818	0.246

表 6: 訓練データ追加による性能比較（DNN7）

訓練データ	acc	pr(X)	rc(X)	f(X)	pr(X+T)	rc(X+T)	f(X+T)	mse
rest1046 のみ	0.474	0.358	**0.924**	0.516	0.745	0.890	0.811	0.223
rest1046+自動評価	0.487	0.434	0.749	0.550	0.746	**0.926**	**0.827**	0.208
rest1046+PDB	0.468	0.414	0.709	0.523	0.735	0.923	0.818	0.215
rest1046+自動評価+PDB	**0.510**	**0.469**	0.673	**0.553**	**0.755**	0.899	0.821	**0.202**

比較すると，いずれの f 値をモデル選択基準に用いた場合も，DNN5（5000-5000-5000）が最大の f 値を示している．一方，DNN6,7 は開発データに対する f 値では DNN5 よりも大きい値を示していた（f 値（X+T) : DNN5 0.841, DNN6 0.857, DNN7 0.855）．テストデータの結果と合わせて考えると，DNN6,7 は開発データに過学習していた一方，DNN5 は，DNN6,7 に比べて層が深すぎず，学習が適切に行われたものと考えられる．

3.2 特徴量の比較

特徴量の比較に用いる学習器として，開発データ上で高い f 値を示していた DNN7 を利用する．DNN7 の結果からは，特徴量の組み合わせによる振る舞いを調べることができる．DNN7 のエポック数の決定基準には，前節でモデル選択基準を開発データ上の f 値（X+T）にしても f 値（X) の性能が低下しなかったことから，開発データ上の f 値（X+T）を用いる．

結果を表 5 に示す．特徴量の追加によって，単語のみを利用した場合よりも，f 値がやや低下している．個々の特徴量を見ていくと，単語，パープレキシティ，直前発話との類似度が，いずれの f 値の向上にも寄与している．一方で単語クラス，自動評価値，パーソナリティ質問フラグはいずれの f 値も下げている．このうち，単語クラスとパーソナリティ質問フラグは，この特徴が性能を向上する場合と低下させる場合とが混在しており，これらの切り分けができていないことが理由と考えられる．また，自動評価値は，推定精度が不十分であったことが原因の一つと考えられる．

3.3 訓練データ追加による影響比較

ここまでの分析では，rest1046 のみを検出器の訓練に利用している．本節では，訓練データに 2.3 節で述べた各データを追加した場合の影響を調べる．比較は DNN7 で行う．結果を表 6 に示す．一方，DNN はいずれのデータ追加においても，性能が改善されている．DNN はもともと過学習の傾向が見られていたため，データの追加によって過学習が緩和されたと予想される．また特に，少数の入力文に多数の応答文が対応している，自動評価データを追加した場合の上昇が大きい．集中的に入力と応答の対応関係を学習することが，検出器の性能向上に有用であったと考えられる．

3.4 チャレンジタスクへの提出データ

チャレンジタスクへ提出したデータは，提出段階までで分析できていた中で，最も高い f 値（X）と f 値（X+T）を示した 2 モデル（DNN7，全特徴量，自動評価データ･PDB データ追加）を利用して生成した．チャ

表 7: 評価データにおける結果

学習器	acc	pr(X)	rc(X)	f(X)	pr(X+T)	rc(X+T)	f(X+T)	mse
提出 1	0**.504**	**0.426**	0.505	0.462	0.789	0.670	0.725	**0.202**
提出 2	0.504	0.415	**0.768**	**0.539**	**0.760**	0.836	0.796	0.203
提出 3	0.469	0.426	0.505	0.462	0.753	**0.848**	**0.798**	0.208

レンジタスクへは 3 通りのデータを提出可能であったため，以下の 3 通りのモデルからデータを生成した．

提出 1 f 値（X）で選択したモデル．開発データ上の f 値 (X)=0.557, f 値 (X+T)=0.762

提出 2 f 値（X+T）で選択したモデル．開発データ上の f 値 (X)=0.508, f 値 (X+T)=0.820

提出 3 提出 1 と 2 を組み合わせたモデル（提出 1 が×を付与したものを×，提出 2 が×か△を付与したものを△）．開発データ上の f 値 (X)=0.557, f 値 (X+T)=0.820

結果を表 7 に示す．提出 1 は，f 値 (X) が開発データ上の値よりもかなり低かった．この結果は，3.1 節の結果とも合致しており，より Recall を高める f 値（X+T）がモデルの選択基準として適切であったことを示している．一方，提出 2 は，f 値 (X+T) がおおむね開発データ上の値に近く，かつ f 値（X) が開発データ上の値よりもやや高かった．しかし，提出 2 は表 6 の最下段と設定が同一であるものの，それよりもやや低い f 値を示しており，学習時のモデル選択の重要性が示唆されている．データを混合した提出 3 は，提出 1 の f 値（X) が低かったため，低い f 値（X) を示していた．

4 結論

本稿では，雑談対話の破綻検出に有用なアルゴリズム，特徴量，データについて分析した．分析の結果，アルゴリズムについては，層・ニューロン数を増やした DNN が SVM を大きく上回っていたことが示された．ただし，やや過学習している傾向が見られたため，層ごとに学習する pre-train の導入によって，さらに性能が向上すると考えられる．特徴量については，検出対象とその直前の発話文に含まれる単語のみで十分に高い f 値を示しており，加えてパープレキシティや直前発話との類似度を追加することで，さらに改善できる可能性が示唆された．こちらについても，pre-train などで適切な最適化を進めることで，個々の特徴量の性質を明らかにしていきたい．加えて，今回導入した特徴量では，どのような話題展開が自然かについて，十分には扱えていない．そのため，人が見ればすぐにわかるような破綻を検出できない場合があった．話題間の距離や，典型的な展開パターンを利用することで，破綻検出性能を向上させていきたい．

参考文献

[Higashinaka 14] Higashinaka, R., Imamura, K., Meguro, T., Miyazaki, C., Kobayashi, N., Sugiyama, H., Hirano, T., Makino, T., and Matsuo, Y.: Towards an open-domain conversational system fully based on natural language processing, in *Proceedings of the 25th International Conference on Computational Linguistics*, pp. 928–939 (2014)

[Meguro 10] Meguro, T., Higashinaka, R., Minami, Y., and Dohsaka, K.: Controlling Listening-oriented Dialogue using Partially Observable Markov Decision Processes, in *Proceedings of the 23rd International Conference on Computational Linguistics*, pp. 761–769 (2010)

[Ritter 11] Ritter, A., Cherry, C., and Dolan, W.: Data-Driven Response Generation in Social Media, in *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pp. 583–593 (2011)

[Sugiyama 14] Sugiyama, H., Meguro, T., and Higashinaka, R.: Large-scale Collection and Analysis of Personal Question-answer Pairs for Conversational Agents, in *Proceedings of Intelligent Virtual Agents*, pp. 420–433 (2014)

[Wong 12] Wong, W., Cavedon, L., Thangarajah, J., and Padgham, L.: Strategies for Mixed-Initiative Conversation Management using Question-Answer Pairs, in *Proceedings of the 24th International Conference on Computational Linguistics*, pp. 2821–2834 (2012)

[杉山 14a] 杉山弘晃, 目黒豊美, 東中竜一郎：大規模マルチリファレンスに基づく雑談対話システムの自動評価に向けた実験的検討, 第 70 回 言語・音声理解と対話処理研究会（SIG-SLUD）, pp. 1–6 (2014)

[杉山 14b] 杉山弘晃, 目黒豊美, 東中竜一郎, 南泰浩：対話システムのパーソナリティを問う質問に対する発話生成, 言語・音声理解と対話処理研究会 (SIG-SLUD-B303), pp. 33–38 (2014)

[大西 14] 大西可奈子, 吉村健：コンピュータとの自然な会話を実現する雑談対話技術, NTT DoCoMo テクニカル・ジャーナル, Vol. 21, No. 4, pp. 17–21 (2014)

[東中 14a] 東中竜一郎：雑談対話システムに向けた取り組み, 第 70 回言語・音声理解と対話処理研究会（SIG-SLUD） (2014)

[東中 14b] 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, 人工知能学会 第 72 回 言語・音声理解と対話処理研究会, pp. 45–50 (2014)

[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)