人工知能学会研究会資料
SIG-SLUD-B402-08

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

Chat dialogue collection and dialogue breakdown annotation in the dialogue task of Project Next NLP

東中 竜一郎[1]　船越 孝太郎[2]
Ryuichiro Higashinaka[1]　Kotaro Funakoshi[2]

[1] NTT メディアインテリジェンス研究所
[1] NTT Media Intelligence Laboratories
[2] (株) ホンダ・リサーチ・インスティチュート・ジャパン
[2] Honda Research Institute Japan Co.，Ltd.

Abstract: Project Next NLP is a project to perform error analyses in the field of natural language processing. In the dialogue task of the project, we are focusing on the error analysis of dialogue systems. This paper describes the ongoing effort of the dialogue task, including how the participants of the task collected chat dialogues by talking to the dialogue system based on NTT Docomo's chat API and how dialogue breakdowns are annotated for error analysis.

1　はじめに

Project Next NLP[1] と呼ばれる自然言語処理分野のエラー分析のプロジェクトが立ち上がっている．このプロジェクトの中で我々は「対話タスク」としての取り組みを始めている．対話タスクは，対話システム研究に関わるタスク参加者が，対話システムのエラーを共同で分析することを目的としたタスクである．本稿の執筆時点で，大学・企業を含む 15 の拠点（表 1 参照）が本タスクに参加している．

対話システムにおいて，エラーを共同で分析することは簡単なことではない．なぜなら，対話システムは多種多様な要素技術・モジュールで構成されるシステムであり，また，その構成方法・採用する技術もシステム毎に大きく異なる．そのため，参加者間で問題を共有・議論することは非常に困難である．

そこで，我々は，参加者間で問題を共有・議論しやすくするために「対話の破綻（文脈上不適当なシステムの応答）を検出する技術（対話破綻検出）」に焦点を絞り，その技術に関するエラー分析を行うことにした．すなわち，特定の対話システムの内部に立ち入る事は避け，その入出力である表層発話だけを対象とすることにした．これにより，システムの種類・構成に関わらず，多くの対話システム研究者・開発者にとって有益な議論がしやすい．

対話破綻検出は将来の対話システムにとって有用な技術となり得る．たとえば，システムがある発話をすると対話が破綻するという可能性を事前に推定できれば，それが回避できる可能性が高まり，よりよい対話システムにつながる．また，たとえ不適当な応答を防ぐ事ができなくても，その後のユーザの反応から対話の破綻が検出できれば，対話の話題を変えるなど，エラーからの回復戦略を取ることも可能となる．

対話タスクは以下のステップで進めている．

1. 破綻を含む対話データを整備する．
2. 整備したデータを分析し，対話の破綻の類型とその原因・対処法を考察する．
3. 整備したデータをもとに複数の破綻検出手法を並行して研究・開発する．
4. 開発した検出器を持ち寄り，破綻検出のエラーを分析し，ボトルネックを同定する．

ステップ 1 は，対話データの収集に関するものである．人間と対話システムの対話で研究に自由に使って良いものは少ない．そのため，今回独自に作成することにした．本稿執筆時点で我々はステップ 1 の段階にあり，後述する予備的なアノテーションを行ったことによりステップ 2 の入り口にさしかかっているところである．

本稿では，タスク参加者が収集した対話データおよびそのデータに対する初期アノテーションの分析について述べる．

2　対話データの収集

対話には，大きくタスク指向型対話と非タスク指向型対話（雑談対話，チャットとも呼ばれる）がある．本タ

参加拠点	秋田県立大学，デンソーアイティーラボラトリ，広島市立大学，ホンダ・リサーチ・インスティチュート・ジャパン，京都工芸繊維大学，京都大学，名古屋工業大学，奈良先端科学技術大学院大学，NTT，大阪大学，首都大学東京，東芝，豊橋技術科学大学，Yahoo!，山形大学
参加者数	32
アドバイザ	篠田浩一先生（東京工業大学），小林哲則先生（早稲田大学）

表 1: 対話タスクの参加状況

スクでは，対話の破綻を網羅的に分析するために，両方の種類の対話において，対話データを収集し，対話破綻のアノテーションを行うことにした．

タスク指向型対話については，既存の人・人あるいは人・模擬システム間の対話を，機械的にシャッフルする処理を行い，人工的に破綻データを生成することにした．非タスク指向型対話については，既存の雑談対話システムを用いて，10,000 システム発話を目標に，人・システム間の雑談データを収集し，そのデータに対し破綻箇所のアノテーションを行うことにした．

2.1 タスク指向型対話データ

名古屋大学武田一哉先生のご厚意により，「名古屋大学 CIAIR 車内音声データベース（以降，CIAIR-ICSD）」から 30 人，60 対話の書き起こしデータを利用する許可を頂いた．

CIAIR-ICSD は，6 つのデータセットからなる．本データにおいて，実験参加者は，人・オペレータ対話（dl），人・模擬システム対話（wz），人・システム対話（lg）の 3 種類の対話を行っている．今回，30 人の実験参加者のそれぞれについて dl と wz の 2 対話ずつを抽出し，計 60 対話を利用することにした．以下は，dl データの抜粋である．ここで，O はオペレータ，D はドライバを表す．

O: はいどうぞ
D: はい
D: ああファーストフードのお店を探してるんだけど
D: どうしようかな
O: はいこの近くですとマクドナルドミスタードーナツロッテリアがございますが
D: じゃあマクドナルドでお願いします
O: はいそれではマクドナルドまでご案内いたします
O: 次の次の交差点二つ目の交差点を左です

現在，破綻対話箇所を自動生成することの妥当性などを議論中である．破綻箇所を生成した後，対話破綻箇所の検出技術を評価するためのデータとして利用していく予定である．

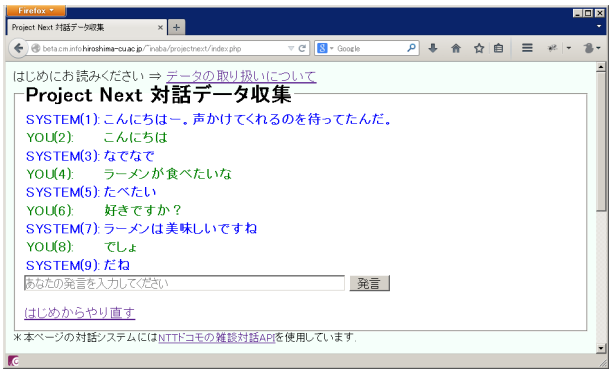

図 1: 雑談対話収集サイトの画面

2.2 非タスク指向型対話データ

NTT ドコモが一般公開している雑談対話 API[2, 3] を用いた雑談対話システムと対話が可能な雑談対話収集サイト [4] を，拠点の一つである広島市立大学内の Web サーバに設置した．このサイトでは，ユーザが 10 発話を入力すると対話が終了し，対話ログが出力されるようになっている．雑談対話収集サイトのスクリーンショットは図 1 の通りである．

各拠点のユーザにこの雑談システムと対話をしてもらい，全部で 1,146 対話を収集した．対話をする際には，付録にある「対話ガイドライン（10 箇条）」に沿うように教示をした．これは，現状の雑談対話システム技術が成熟していないこともあって，破綻の対話ばかり収集されてしまわないためである．話者毎の対話数は 1 から 40 までばらつきがあるが，全体の対話者は 100 名を超えている．表 2 は収集された雑談対話データの統計情報である．収集した雑談対話の一例を以下に示す．S はシステム発話，U はユーザ発話を表す．

S: こんにちは！声かけてくれるのを待ってたんだ。
U: 今日は最高気温 36 度だって。暑いねえ
S: 最高気温は１７度が予想されます??
U: いやいや猛暑ですよ
S: 猛暑に備えてますか？
U: うん、日傘さしたり、水飲んだり、クーラー入れたりしてる
S: クーラーは快適ですね
U: うん、会社だと朝はつけられないから死にそう
S: 会社は冷房が効きますよねえ
U: 昼間は効くね。ここはパソコンがいっぱいあるからないと大変
S: 動きが悪いです
U: そうなの？こっちは快適だよ
S: パソコンは使い勝手が悪いです

対話数		1,146
話者数		116
	ユーザ	システム
発話数	11,460	12,606
異なり発話数	10,452	7,777
単語数（形態素数）	86,367	76,235
異なり単語数	6,262	5,076

表 2: 収集された雑談対話データの統計情報．形態素解析には MeCab の version 0.996 を使用した．システム発話数は 12,606 となっているが，これは最初のプロンプト発話の 1,146 とユーザへの返答の 11,460 を合わせた値である．

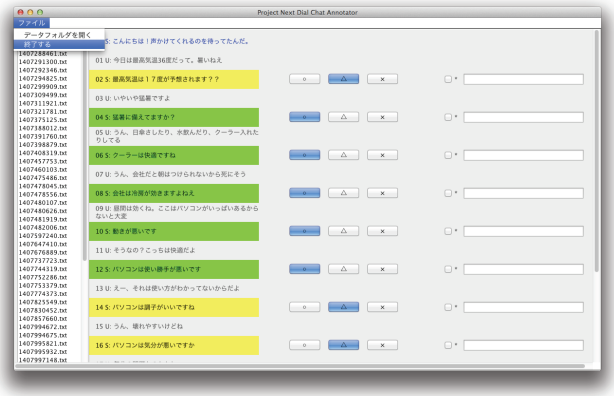

図 2: 雑談データ用破綻アノテーションツール

3　対話破綻アノテーション

収集した雑談対話データに対して対話破綻の初期アノテーションを行った．まず，1,146 対話からランダムに 100 対話を抽出し，全 100 対話のアノテーションを行える 24 人によってアノテーションを行った．このアノテーションの目的は，残りの 1,046 対話に対して，1 対話あたり何人のアノテータを割り当てるのが妥当かを検討するためである．ここで作成したデータのことを **init100** データと呼ぶ．

アノテーションについては，どのようなエラーがあるのかを網羅的に分析したいという目的に鑑み，トップダウンな破綻の分類は示さず，直感に従って ◯・×・△ の 3 分類でアノテーションするように指示した．それぞれの意味は以下の通りである．

◯ **破綻ではない** 当該システム発話のあと対話を問題無く継続できる．

△ **破綻と言い切れないが，違和感を感じる発話** 当該システム発話のあと対話をスムーズに継続することが困難．

◯	△	×
59.2% (14212)	22.2% (5322)	18.6% (4466)

表 3: ◯×△ の発生割合（発生数）

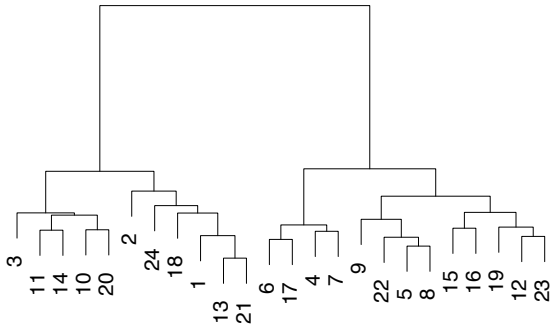

図 3: アノテータのクラスタリング結果

× **あきらかにおかしいと思う発話．破綻** 当該システム発話のあと対話を継続することが困難．

アノテーションには，図 2 に示す専用のツールを使用した．ツールでは，非文のチェックの他に，各発話に対してコメントを記入できるようになっている．また，先行する文脈にのみに基づいて対話破綻のアノテーションが出来るように，先頭から順に 1 発話アノテーションする毎に，次のユーザ発話とシステム発話が表示されるようになっている．なお，破綻とタグをつけた後の発話をどうアノテーションするかについては，対話の先頭から，破綻とタグ付けされた発話を含むこれまでの文脈を「ありき（与えられたもの）」として，アノテーションするように教示した．

3.1　アノテータ間の一致度の分析

init100 データに対して，24 人のアノテータが付与したラベル ◯，△，× の割合を表 3 に示す．24 人のアノテータ間の一致の程度を測るために Fleiss の κ を算出すると，0.276 であった．この値の解釈は「ランダムではないが，よく一致しているともいえない」とするのが妥当である．△ を × に含めて，2 値のアノテーションとして計算すると，0.396 とやや一致の具合が高まる．△ を ◯ に含めると κ は 0.277 にしか改善されないため，△ は × により近いことが分かる．

24 人のアノテータを Cohen の κ 値をもとに Ward 法で階層クラスタリングを行うと，図 3 のようになった．距離の定義やクラスタリングの手法を変えると，2 つのクラスタの中でのまとまり方は細かく変わるものの，大きな 2 つのクラスタ間での移動はほとんど見られなかった．表 4 に示す 24 人のアノテータの分布を見ると，◯ をつける傾向の大小で，前述の 2 クラスタが分かれていることが見て取れる．2 つのクラスタの中

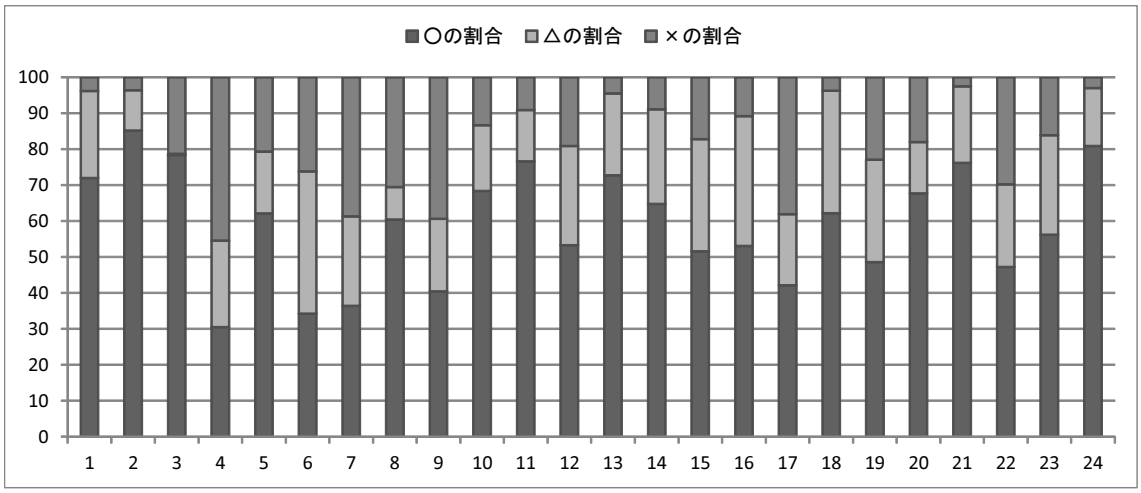

図 4: アノテータ毎の ◯△× を付与した割合．横軸はアノテータ ID.

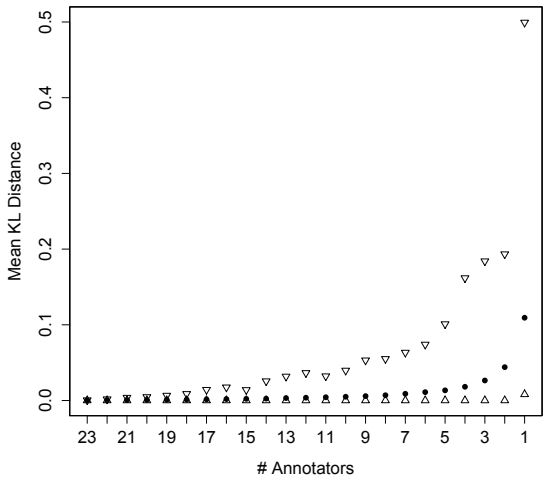

図 5: 24 人のラベル分布とランダムサンプリングした N 人のラベル分布の異なり

での κ 値を求めると，それぞれ 0.414（11 人）と 0.474（13 人）であった．

24 人のアノテータからランダムに N 人を選び出したとき，ラベルの分布がどれだけ全体の分布から離れているのかを表したグラフを図 5 に示す．横軸は N の数で，縦軸は Kullback-Leibler divergence の対称平均の値である．黒丸が 1,000 回のサンプリングした際の平均値を示す．下向き三角は 1,000 回中の最大値，上向き三角は 1,000 回中の最小値を表す．アノテータが 1 人から 2 人になる段階で，平均値からの乖離は半分近く縮まり，あとは，なだらかに 24 人の分布に近寄っていくことが分かる．

◯ の数と △ と × の数が半分ずつアノテーションされた発話の分析も行った．すると，表面的には間違ってるとは言えないけれども，違和感がありそうな応答が多かった．「こういうことは普通ここでは言わない」という印象は持つものの，なぜそれを言わないかの説明が難しいものが多い．また，発言の意図が読みにくそうな発話，ユーザの話を聞いていないと感じられる発話，社会的に問題がある発話，ふざけているようにも見える発話なども判断が割れていた．これらの判定を分析することは，対話における認知の研究として学術的に面白い課題となり得る．以下は，評価が割れた発話（下線部）を含む対話の例である．

U: 他に好きな楽器はありますか？
S: サックスがいいですね
U: サックスは木管楽器なんですよね
S: 相性がいいですね
U: 台風は大丈夫でしたか
S: 台風は必要です
U: 雨は必要ですよね
S: 朝から雨が降るのです

これは発話の意図が読みにくいため，「それがどうしたのか」という印象を受ける発話である．

U: 夏って感じがします
S: 海は騒がしいですよねえ
U: 海に行くと盛り上がりますもんね
S: 海は静かなんですよねー
U: 騒がしいのか静かなのかどっちですか
S: 海はサイコーですよねえ
U: サイコーですねえ

これは人によっては，質問に答えずにごまかしているように感じられるだろう発話である．

3.2 コメントに基づく破綻の分類

init100 データ中の一部の対話について，アノテータによって対話破綻箇所に付与されたコメントを人手で分類したところ，大きく「同じ内容の繰り返し」「矛盾した内容」「発話として唐突」「ユーザ発言内容の無視」「質問に答えていない」に分類できることが分かった．

同じ内容の繰り返し 表層としては少し異なっていたとしても内容として同じものを繰り返す場合，破綻とアノテーションされることが多かった．たとえば,「美味しいですね」「いいですね」などと同じような発話を繰り返す場合である．

矛盾した内容 システム発話間で矛盾が見られる場合は破綻とされることが多かった．たとえば,「イチゴが好き」という発言の直後に「リンゴが好き」と発言するなど，一貫性を欠く発話は問題視された．

発話として唐突 「おはようございます」に対して「明けましておめでとうございます」のように，文脈とは関係のない発言を突然行うことがあり，このような発話は破綻とされていた．

ユーザ発言内容の無視 対話はお互いが協調して進めていくものであるので，ユーザ発話を全く受けずにシステムが発話を行った場合には対話の破綻とみなされることが多かった．たとえば，旅行の話をしていて「車で行きましょう」とユーザが話しかけたのに「車はかっこいいですね」と車そのものについて言及したりする場合である．

質問に答えていない ユーザ発言内容の無視に近いが，特に質問に答えていないものが破綻とされていた．たとえば,「チワワは欲しいですね」とシステムが話し，それに応じてユーザが「飼う予定はあるの？」と質問したが，システムは「チワワはいいらしいですよ」と答えたような場合である．

上記以外にも口調の唐突な変化などが，問題のある現象として観察された．残りの対話についてのコメントについても分析を行い，より詳細に何が破綻と思われるかについての知見を得たい．対話破綻アノテーションを元に自動的に対話破綻を検出するプログラムを作成することも重要であるが，もう一方で，対話の破綻を類型化することも重要であり，これは対話システムのエラー分析そのものである．

3.3 今後のアノテーションの進め方

init100 データに対するアノテーション結果について，タスク参加者で議論を行った結果，残りの 1,046 対話のアノテーションについては，1 対話につき 2 人で実施するという結論に至った．2 名とした理由は以下の通りである．

- 人的・経済的コストの面から，アノテーションにかかる作業量は最小限が望ましい．
- アノテーションのコストを最小化できるのは 1 名でアノテーションを行う場合であるが，この場合，アノテータ間の揺れのために，破綻とされるべき発話が見逃されてしまう可能性がある．よって，複数名が望ましい．
- 前述の分析でアノテータは大きく二つのクラスタに分かれることが分かっている．これらの 2 つのクラスタから 1 名ずつ割り当てることで，見逃しを最も効率的に減らせる可能性がある．

残りのアノテーションは年内を目標に行い，その後，破綻箇所の分析をして，対話破綻の類型を作り上げていく予定である．破綻の分類を考える際にはアノテーション時のコメントが重要な情報源となるが，全ての箇所にコメントすることはアノテータの負担が大きいので，予め取り決めた一定数の対話についてはコメントを義務づける予定である．

このようにして作成される雑談対話データは，人工的に破綻箇所を作成した課題指向型対話データとともに，タスク参加者で共有し，各拠点において，対話破綻検出プログラム作成に役立てていく予定である．加えて，今回作成したデータは，対話システムにおける評価型ワークショップの共有データとしても整備していく予定である．

4 関連研究

タスク指向型の音声対話対話システムの文脈では，音声認識，発話理解，対話管理などの各モジュールから得られる特徴量から対話に破綻が起きているかどうかを判定する手法がいくつか提案されている．たとえば，Walker ら [5] や Herm ら [6] は，コールセンタにおける通話について，問題が起こっているかどうかを数ターンで判定する判定器を機械学習の手法で構築している．対話中のユーザの満足度の遷移を推定する研究もされている [7]．これらは雑談対話を扱ってはいないが，目的意識は本稿での取り組みと近い．

雑談対話においては，Chai らがユーザの対話行為の系列の情報を用いて，問題のある質問応答ペアかどうかの判別を行っている [8]．Xiang らは，対話行為に加え，感情の系列を用いることで，雑談対話における問題発話の検出を行っている [9]．Higashinaka らも，雑談対話システムの発話の結束性をさまざまな素性から推定する手法を提案している [10]．しかしながら，これらの研究は精度がいまだ高いとは言えず，また，対話破綻の類型化なども行われていない．対話タスクで

は，エラー分析を詳細に行うことで，対話破綻の原因を明らかにし，高精度な破綻検出を実現したいと考えている．

5 まとめ

本稿では，Project Next NLP における対話タスクの目標やステップ，そして，これまでに行った雑談対話データの収集と対話破綻アノテーションについて報告した．init100 データの初期分析により，ユーザによって対話破綻のとらえ方が大きく違うことが明らかになった．また，アノテータによるコメントを分析することで，初期的な類型化も行った．

対話破綻のアノテーションがされたデータが多くなれば，そのデータを用いて様々な機械学習の手法が試されるようになるし，また，対話破綻の類型化が進めば，より破綻しない対話システムに向けた指針が明らかになっていくだろう．対話タスクの営みは，現状で特定の対話システムのエラー分析であるが，同じ目的意識を共有しにくい対話システムという分野において，一定の方向性を与える重要なものであり，これを機に，国内の対話システムコミュニティを活性化していきたい．

なお，本プロジェクトで作成する対話データおよびアノテーションデータは，当面は，プロジェクトに参加するメンバーに限定して配布するが，一定期間の後，一般に公開する予定である．ただし，CIAIR-ICSD については，公開予定のデータはテキストだけで，音声等については配布元から別途入手が必要であるので注意されたい．エラー分析やアノテーションにご協力いただける対話タスクのメンバーも随時募集中である．

謝辞

対話データの収集にご協力下さったみなさま，対話破綻アノテーションにご協力頂いた拠点参加者のみなさま，タスク指向型対話データをご提供頂いた名古屋大学の武田一哉先生に感謝いたします．また，本稿のドラフトに有益なコメントを頂いた，中野幹生氏，荒木雅弘氏，駒谷和範氏に感謝いたします．

参考文献

[1] Project Next NLP. https://sites.google.com/site/projectnextnlp/.

[2] 雑談対話 API. https://www.nttdocomo.co.jp/service/developer/smart_phone/analysis/chat/.

[3] 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.

[4] 雑談対話収集サイト． http://beta.cm.info.hiroshima-cu.ac.jp/~inaba/projectnext/.

[5] Marilyn Walker, Irene Langkilde, Jerry Wright, Allen Gorin, and Diane Litman. Learning to predict problematic situations in a spoken dialogue system: Experiments with How May I Help You? In *Proc. NAACL*, pp. 210–217, 2000.

[6] Ota Herm, Alexander Schmitt, and Jackson Liscombe. When calls go wrong: How to detect problematic calls based on log-files and emotions? In *Proc. Interspeech*, 2008.

[7] Alexander Schmitt, Benjamin Schatz, and Wolfgang Minker. Modeling and predicting quality in spoken human-computer interaction. In *Proc. SIGDIAL*, pp. 173–184, 2011.

[8] Joyce Y Chai, Chen Zhang, and Tyler Baldwin. Towards conversational QA: automatic identification of problematic situations and user intent. In *Proc. COLING/ACL*, pp. 57–64, 2006.

[9] Yang Xiang, Yaoyun Zhang, Xiaoqiang Zhou, Xiaolong Wang, and Yang Qin. Problematic situation analysis and automatic recognition for chinese online conversational system. In *Proc. CLP*, pp. 43–51, 2014.

[10] Ryuichiro Higashinaka, Toyomi Meguro, Kenji Imamura, Hiroaki Sugiyama, Toshiro Makino, and Yoshihiro Matsuo. Evaluating coherence in open domain conversational systems. In *Proc. Interspeech*, pp. 130–133, 2014.

付録：対話ガイドライン (10 箇条)

1. たまたま待合室や飛行機などで隣り合った見知らぬ人と話すイメージで対話しましょう．
2. システムの発話はなるべく好意的に解釈し，対話を続けるように努力しましょう．
3. 対話毎に新しい気持ちで話しましょう．
4. 自分から話題を開始しましょう．
5. 自分から話題を変えましょう．
6. 何度も同じことを繰り返さないようにしましょう．
7. 誹謗中傷は控えましょう．
8. 個人情報を入力しないようにしましょう．
9. 日本語で入力しましょう．
10. 上記を守っていれば，あとは自由です．対話を楽しみましょう．