単語間共起及びキーワード抽出を用いたルールに基づく対話破綻自動検出器の構築と評価手法の検討

Dialogue Breakdown Detectors using Rules with Keyword Extraction and Word Co-occurences Examining Evaluation Method

谷口諒輔 [1]　　狩野芳伸 [1]

Ryosuke TANIGUCHI[1],　　Yoshinobu KANO[1,]

[1] 静岡大学 情報学部

1. はじめに

Project Next NLP 対話タスク[1]では、NTT ドコモ社が提供する雑談対話 API[2]を使用した中で獲得されたシステム発話とユーザー発話、またはそれら一連の会話について、複数のアノテーターによって付与された破綻ラベルを元に、対話システムのエラーを分析することを目的としている。しかし、そのような方法では人的コストが嵩むばかりか、それを基に機械学習を利用するには圧倒的にデータ量が足りていないのが現状である。

対話破綻検出チャレンジは、人間に代わりシステムが自動的に破綻を検出を技術の確立を目指した評価型ワークショップである。このような破綻検出器の開発は、対話システムが自発的に破綻したことを認識することができれば、リカバリを試みことが可能となる[3]。

本チャレンジで、我々は破綻ラベルが付与された表層的な発話内容に着目し、ヒューリスティックに導かれた独自の破綻ルールを制定、それらを組み合わせた手法を用い、学習データを必要としない極力単純化された対話破綻検出器の実現を試みた[4]。本論文では、チャレンジで制作した破綻検出器の説明に加え、日本語 Google n-gram コーパス[5]によって計算した単語間共起スコアを利用し、単文レベルでの破綻検出が可能かを検討する。

2. 提案手法

2.1 破綻事例の観察

対話破綻検出チャレンジで配布された、ユーザ（人間）とシステムの対話において、システムは前段階におけるユーザ発話もしくは自身の発話内容から部分的に単語を抽出し、その単語を反復して次ターンの発話内容に利用するケースが多く存在する。以下は、雑談破綻チャレンジにおいて配布された雑談対話における一例を部分的に抜粋し示したものである。S はシステムを、U はユーザを表す。

S:予防は完璧ですね
U:はい。あなたは何か熱中症対策をしていますか？
S:予防が肝心です
U:そういえば、梅雨が明けましたね。
S:日頃から予防を心掛けるかもしれない

このように、各発話の話題の中心はこの繰り返されて使用される単語によって決まることが多いため、会話文中のシステムとユーザの発話の前後、またはシステムの発話における前後についてキーワードを抽出できれば、焦点の遷移をある程度把握することができる。

以下では、配布された init100 から 20 対話、rest1046 より 30 対話を、破綻ルールの策定の観察の対象として無作為に選出し、使用した。これにより対話破綻と見受けられる点について、その内容と例を抜粋して紹介する。

(1) **ユーザの質問文に対して、適切な返答をシステムが行うことができない。**
U:昼ごはんは何を食べましたか
S:ごはんはあったかいです

(2) **システムの発話について、非文ではないが文章として不適切なとき。**
S:夕食がいってないかもです
U:日本語として変です
S:日本語は早いです

(3) **システムが、自身の発話した中から特定のワードを用い、再度発話したとき。**
S:こんばんは。夏といえばスイカだね。
U:そうだね。でもまだ食べてないや。
S:スイカで買い物できてるのですか？

(4) **ユーザの発話の直後のシステムの発話について、動作主は同一にも関わらず、助詞が変換されているとき。**
U:そうかな？私は旦那さんと買い物するよ。
S 旦那が行きますか？

(5) **ユーザの発話前後のシステムの発話について、そのどちらにも特定のキーワードが含まれているにも関わらず、ユーザ発話にはその特定のキーワードが含まれていないとき。**
S:エボラ出血熱を引き起こしますよねー
U:何が原因なんでしたっけ
S:エボラ出血熱を引き起こすらしいです

(6) **システムの発話における「〜です・ます」または「〜だ・である」といった、口調の不一致**

S:行楽シーズンにはいりますか？
U:ええ
S:どうした

(7) **システムの発話における時勢の不一致**
S:こんにちはー。ちょうど退屈してたんだ。
U:こんにちは
S:おはよ

(8) **システムの発話において、過去の会話中で一度断定された内容に関して、再度質問形式でユーザに発話したとき。**
S:プールは水が生ぬるくないですよ
U:気温によるかな
S:水が生ぬるいとかですか？

これらのうち(6)、(7)については、破綻とみなすかどうかについてアノテーターによって異なることから、主観的な判断が強いと推測されるため、ルールとしては不適切である。(2)については Google n-gram を用いた単語間の共起確率を利用した手法でアプローチを試みる。

2.2 破綻ルールの策定

前述した項目のうち、対象とするものを破綻ルールとして策定する。内容的に重複しているところもあるため、下記に示される 3 点として一般化し、ルールとして策定する。

(Ⅰ)ユーザが話題の転換の後、そのターニングポイント前のワードをシステムが発話したとき。言い換えるに、システム 1、ユーザ、システム 2 の順の発話内容の内、システム 1 とシステム 2 には同一のキーワードが含まれているにもかかわらず、ユーザにはそれに一致するキーワードが含まれていない場合に破綻ラベルを付与する。

(Ⅱ)ユーザの質問発話直後のシステムの発話に破綻ラベルを付与する。

(Ⅲ)システムの質問発話について破綻ラベルを付与する。

ただし、(Ⅱ)、(Ⅲ)に関して、質問発話であるという判断基準は、各発話の文末にクエスチョンマークが含まれているかのみで判断する。(2)に該当するような、不適当な日本語の使用の検出は Google n-gram を使用し、得られたスコアを元に破綻か否かを判断する。

実験では、前述した 3 つのルールを次のように適用し、ラベルを付与した。ラベルは O（破綻ではない）、T（破綻とは言い切れないが違和感を感じる発話）、X（明らかにおかしいと思われる発話）の三種類が与えられる。

run1:(Ⅰ)(Ⅱ)(Ⅲ)のルールを OR で接続し判定する。
3 つ全ての、または 3 つのうち 2 つのルールに該当する場合は破綻ラベル X を付与する。ただし 2 つのルールに該当する場合は、3 つのルール適用時よりも破綻可能性を軽減させる。1 つだけ該当する場合はラベル T を付与する。

run2:(Ⅰ)(Ⅱ)(Ⅲ)のルールを AND で接続し判定する。
いずれかのルールについて一つでも該当するものがあれば破綻ラベル X を付与する。

run3: (Ⅰ)(Ⅱ)(Ⅲ)のルールを単体で運用した際に、最も判定正答率が高かった(Ⅲ)のみを使用、該当のものに破綻ラベル X を付与する。

キーワードの抽出については、Java で実装されたオープンソースの日本語形態素解析器 kuromoji[6]と、kuromoji のユーザ辞書として Wikipedia データを用いた。我々の手法では、形態素解析時に、ユーザ辞書に登録された文字列については最長文字列マッチ相当によって品詞分解されるため、通常の形態素解析結果を利用した場合と違い強制的に単語が区切られることになる[7]。またユーザ辞書に登録された単語が検出された場合、Wikipedia という属性値を与えた。本検出器では、この属性値が与えられた単語をキーワードとして扱い、これがシステムとユーザの各発話の焦点であると考える。ただし、Wikipedia データ内には挨拶、ひらがな 2 文字の単語、また文末の助動詞などの本タスクの目的にとっては有害なエントリがある。これらのエントリについては、一度与えられた対話データについて形態素解析を行い、必要に応じて目視で辞書から除外した。

3.結果と考察

3.1 ルールベースの手法

Accuracy（全ラベルの一致率）について run1 から run3 を俯瞰すると、ルールを単体運用した run3 の正答率が高いということを読み取ることができる。しかし、雑談対話データの収集の際の各ラベルの発生数を参考とするに、破綻ではないラベル O をアノテーターは最も多く与えていることから、しきい値を 0 から 1 に近づけ正解アノテーションをラベル O にスライドさせていくことによって、そもそもラベル O を多く与えた run3 の一致率が高いことは必然であり、Accuracy を用いた一概の評価は有意であるとは言えない。これより、以下に記された評価指標について各検出器を検討する。

3.1.1 Recall(再現率)と Precision(精度)

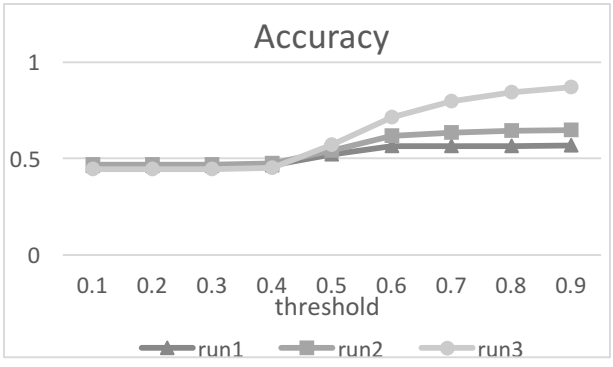

図 1

図 1 より run1、run2 のラベル X と T の評価総数に変化はないため、しきい値の変化に伴う再現率及び精度の推移は一致している。両者の評価結果での最大の差異はラベル X の再現率であり、しきい値が 0.5 のとき、その差は約 50%と run2 の方が網羅性が高いと読み取れる。同時に、精度に関しても run2 が上回っていることから、本来破綻ラベ

ルを付与すべき対話について run1 では読み過ごされていると言える。run3 に関しては評価ラベルの偏りを前述した通り、再現率と精度は共に低い水準となった。

3.1.2 JS ダイバージェンスと MSE

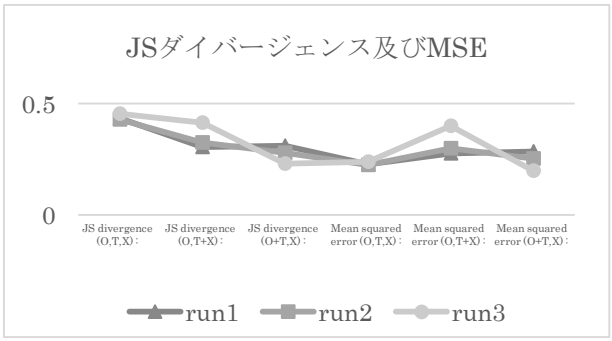

図 2

図 2 より、run1 と run3 は近似した精度評価指標を有しているのに対し、run2 は異なった結果を示していることがわかる。run1、run3 はラベル T が O に近い場所にあるとしているが、run2 では T は X 寄り付与されている。

run1 は 3 つの検出器で唯一 T を付与できるものであるが、単体のルール(Ⅲ)を適用した run3 が近似した理由は以上の二つが考えられる。

① ルール(Ⅲ)で破綻と評価される発話総数が、ルール(Ⅰ)(Ⅱ)よりも相対的に少なかったため。

② ルール(Ⅲ)で破綻と評価される発話が、ルール(Ⅰ)(Ⅱ)に多く内包されているため。

アノテーターによって付与された母集団のラベルが O に偏在していること、また上記の理由から run3 の一致率が上昇した原因を推測できる。

対して run2 は、元来 run1 でラベル T として扱われた発話をすべて破綻と認定しているためラベル X は T と大きく乖離する結果となった。またこの二値的な分類方法は、結果として X と T 間、T と O 間の曖昧さを排除し、しきい値が大きくなるほど、つまり厳格な評価になればなるほど評価値が良くなることにつながったと考える。

3.1.3 F_β スコアによる評価

与えられた対話データに対するアノテーションが全体としてラベル O に偏在していることに加え、複数のルールを論理和として取ることによって精度が下がることは必然的である。機械学習等の手法においては、しきい値や総合値によって両者のバランスを調整することが可能であるが、我々の手法においてはそのような手段を持ち合わせていない。そこで、ラベル O とラベル T の比率を加味した F_β を用いて再評価する。ここで、

$$F_\beta = (1+\beta^2)\frac{precision \cdot recall}{\beta^2 \cdot precision + recall}$$

であり、β の値には母集団のラベルの比率（ラベル X とそれ以外、またはラベル T およびラベル X とそれ以外）を用いた。

表 1

表 1 は run1 におけるしきい値 t を 0.4 から 0.6 まで変えた場合の F_1 値、及び F_β 値の比較結果である。F_β 値を用いた評価手法は、元来の F_1 値と比べても比較的高い数値を、とりわけ X における F_β 値は約 10%以上高い数値を得ていることが表から読み取れる。また図 3 は、t=0.5 における run1 から run2 のそれぞれの F 値を示している。

threshold	0.4	0.5	0.6
F-measure (X) :	0.423567	0.384083	0.268
F-measure Beta (X) :	0.500728	0.52322	0.530381
F-measure (T+X) :	0.59805	0.59805	0.580571
F-measure Beta(T+X) :	0.607295	0.55115	0.519057

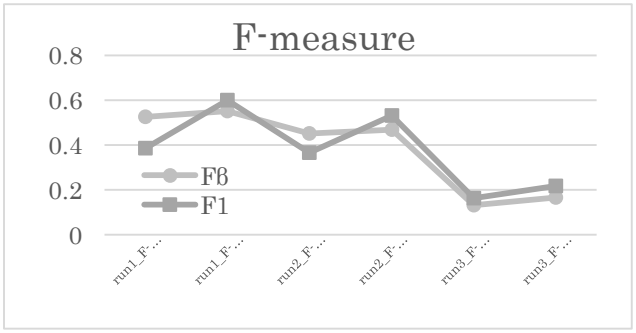

図 3

複数のルールを組み合わせた run1、run2 について、F_β は F_1 と同等かそれ以上の結果を導いている。対して、run3 についてはあまりその効果が見られない。前述の通り run3 は、Accuracy の高さのみに着目し運用されているため、ラベル O を多く付与した破綻検出器である。したがって、ラベル O とラベル X の比率を考慮した F_β でも F_1 との差は小さくなることになる。以上の結果から、F_1 ではなく F_β を用いることで、偏りのあるアノテーション集合に対してより適切な評価ができていると考えられる。

F 値において、我々の破綻検出器は一部を除き上回る結果となった。その一部については、ラベル T の出力を多くしたためであると考えられる。

3.2 単語共起頻度

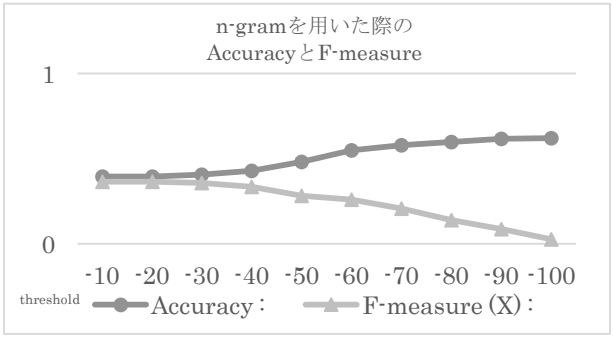

図 4

図 4 は、確率値をしきい値として用い対話データを評価した結果を示している。t 値を小さくすればするほど Accuracy は上がっていき、全ての発話にラベル O を付与したときの値（0.63）に近似していく一方で、F 値は下が

っていくことから、発話内容に関係なくラベル O をいかに多く出力したかという点に結果が左右されていることを読み取ることができる。また実験より、次のような発話例とそのスコアを得ることができた。

S:朝から買い物に出かけてるのですか？ : NaN

S:午後から買い物に出かけてますか？ : -46.6399

(NaN は計算不可能であることを表す)

文末に全角クエスチョンマークが用いられている場合、スコアを計算することができない。しかし上記の一例のように稀に計算可能な発話もあり、その原因は特定できていない。

S:雨/は/心地よい/です/ね:-35.225594

S:心地/が/体/に/良い/です/ね:-48.1727

(文中のスラッシュは形態素分割の位置を表す)

上段の例では 6 単語、下段の例では 7 単語で構成された発話だが、文中の単語数がスコアに大きく依存していることがわかる。また、

S:心地/に/こだわっ/たり/と/か/です:-52.64904

と比較すると、確かに 7 単語で構成されている発話同士であれば、比べられる可能性が高い。

Gooogl n-gram コーパスを用いた単語共起頻度による手法には、以下のような問題点が挙げられる。

(1)Kuromoji による形態素解析結果と n-gram 解析に使用される文字列が一致しない。例えば、Kuromoji では「時代劇」と一括りになっている単語も、Google n-gram では「時代」と「劇」に分割されており、辞書外の文字列と認識される。形態素解析に用いる辞書を Google n-gram に一致させれば解決できる可能性があるが、これは現在運用している Wikipedia ベースのユーザ辞書を利用したキーワード抽出機構と競合する可能性がある。

(2)Google n-gram コーパスに登場しない新出の単語列は計算できない。

(3)文中の単語数にスコアが強く依存し、日本語として不適切な発話であっても発話が短いと有意であると捉えられたり、またその逆のパターンが多く見られる。このため、二値分類的な破綻検出が困難である。

まとめると、Google n-gram を適切に運用するにあたって第一に、与えられた会話データを解析可能な形に変換する必要性がある。とりわけ記号に関しては注意が必要である。またキーワードを取得する形態素と n-gram を計算するための形態素を分けることも考慮する必要があると考える。何よりもスコアの算出方法とその利用については再考の余地がある。各発話中の単語数の大小によってスコアが左右するのであれば、正規化、もしくは雑談対話 API が生成しやすい似通った発話に着眼し、その中で比較検討するといった工夫が必要だろう。

4.まとめ

本稿では対話破綻検出器について、どのような対話破綻の形式があるかを対話データから読み取り、それに基づいた一般化したルールを作成し自動化を試みた。その結果に対して F_β を用いた新たな評価手法の使用と、Google n-gram の利用を検討した。

これまでに開発した破綻検出器は F 値について、ベースラインを上回る結果となった。また、ラベル X とラベル O の比率を加味した F_β 値を用いた評価手法の導入は、新しい側面における評価として価値があるものと考えられるのではないか。ただし、これまでに作成した対話破綻検出器の、精度向上を目的とした Google n-gram の導入であったが、対話データのテキスト形式に十分対応できていなかったこと、また導かれたスコアをどのように運用していくかという点について、改善すべきポイントが残されている。加えて、元来の対話破綻検出器のチューニングについては十分に言及できていない。第一に、各ラベル間を二値的に、厳しく評価してしまうことが挙げられる。この点の解決策については、3 つのルールについて最適化した確率変数の振り分けを行うことであろう。今回導入を試みた Google n-gram と併用することも視野に入れて、最適化することを目指さなければならない。

参考文献

[1] 東中竜一郎, 船越孝太郎, 荒木雅弘, 塚原裕史, 小林優佳, 水上雅博: Project Next NLP 対話タスク：雑談対話データの収集と対話破綻アノテーションおよびその類型化. 自然言語処理におけるエラー分析（兼：Project Next NLP 報告会）, 言語処理学会第 21 回年次大会(NLP2015) (2015)

[2] NTT ドコモ, 雑談対話 API: http://www.nttdocomo.co.jp/service/developer/smart_phone/analysis/chat/

[3] 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, SLUD 研究会 (2015)

[4] 谷口諒輔, 狩野芳伸: キーワード抽出を用いたルールに基づく対話破綻自動検出器の構築, 第 6 回対話システムシンポジウム, SLUD 研究会 (2015)

[5] kuromoji: http://www.atilika.com/ja/products/kuromoji.html

[6] Google n-gram: 工藤拓, 賀沢秀人: Web 日本語 N グラム第 1 版, 言語資源協会発行

[7] 狩野 芳伸: 大学入試センター試験歴史科目の自動解答, 2014 年度人工知能学会全国大会（第 28 回）(2014)