発話ベクトルの差分特徴量を用いた雑談対話システムにおける破綻した話題遷移の検出

豊嶋 章宏[1]　吉野 幸一郎[1,2]　須藤 克仁[1]　中村 哲[1]
[1] 奈良先端科学技術大学院大学 情報科学研究科
[2] 科学技術振興機構 さきがけ
{toyoshima.akihiro.su4, koichiro, sudoh, s-nakamura}@is.naist.jp

1　はじめに

人と会話やコミュニケーションを行うための，雑談対話システムに注目が集まっている．雑談対話システムは，雑談を行うことそのものが目的であり，多様なドメイン間を文脈に応じて遷移する必要がある．この際，会話のドメインをうまく遷移することができず，システムが受け取ったユーザの発話や，会話の文脈に対して不適切な応答を行ってしまう場合がある．このような不適切なシステム応答に由来する話題遷移の失敗を，本研究では破綻した話題遷移と呼ぶ．破綻した話題遷移が生じた結果，ユーザがシステムとの対話に対し，期待の喪失や失望感，疲労，怒り等ネガティブな感情を抱き，対話の継続が困難な状況に陥る場合がある (対話破綻)[5]．雑談対話システムでは，ユーザとのコミュニケーションそのものが目的となっているため，破綻した話題遷移が起きないように対話制御を行い，ユーザの興味や関心を保つことが重要な要素として挙げられる．

そこで本研究では，破綻した話題遷移による対話破綻を検出する手法について取り組む．これにより，破綻した話題遷移が起こりうるシステム発話の抑制が可能となるため，適切な話題追従や話題展開が実現できる．具体的には，ユーザ・システム発話ペアを入力としたときに，それぞれの発話文を発話ベクトルへ変換し，発話ベクトル間の差分情報を基に対話破綻の検出を行う．破綻した話題遷移に限定した対話破綻データを用いて評価実験を行い，明らかに破綻しているような発話に対しては，高い精度で検出できることを検証した．

2　関連研究

2.1　雑談対話システムにおける話題遷移

話題遷移を考慮した雑談対話システムに関する研究として，Higashinaka らの研究 [2] や Tsukahara らの研究 [9] がある．Higashinaka らの研究では，発話文の中心となる話題を示す焦点語を抽出し，これを用いた発話生成を行っている．Tsukahara らの研究では，対話における発話文の役割を示す対話行為や，発話文のトピックを推定した結果を用いた発話生成を行うことで，いくつかの事例に対して適切な話題遷移を実現している．これらの研究では，直前のユーザやシステム自身の発話内容に対して話題が追従できないことや，焦点語の解析を誤った結果，誤った話題展開をしてしまうなどの課題がある．

2.2　対話破綻検出チャレンジ

本研究で扱う対話破綻検出に関するタスクとして，対話破綻検出チャレンジ (DBDC) がある．DBDC(Dialogue Breakdown Detection Challenge) は，雑談対話システムとユーザの対話に対し破綻ラベルを付与し，このデータから対話破綻を検出する手法を開発する評価型ワークショップである [10][3]．用意されている対話データは，それぞれ 21 発話 (内システム発話は 11) で構成され，各システム発話に対して複数人のアノテータが破綻ラベルを付与している (2〜30 人)．付与されているラベルは，破綻していないことを示す”NB”，破綻とは言い切れないが，違和感を感じる発話を示す”PB”, 破綻を示す”B”の三種類である．

DBDC3 において，最も精度が高い Sugiyama らの手法 [7] では，推定対象となるシステム発話と直前のユーザ発話との単語類似度や，各発話の対話行為や文の長さなどを抽出し，発話間の共起情報を特徴量として用いている．さらに，この特徴量を元に，SupportVectorRegressor や ExtraTreesRegressor といった回帰分析手法を，Stacked regression model を用いてアンサンブルすることで，破綻予測を行っている．

精度の高い先行研究の手法を見ると，対話特有の特徴量の設計や，データをいくつかに類型化してモデルを設計すること，いくつかの手法を用いてモデルを設計しアンサンブルするなどの方法が有用であることが考えられる．そこで本研究でも，対話特有の特徴量である発話間の差分情報や，複数の素性を組み合わせてモデルの学習を行い，対話破綻推定に取り組む．

3 発話間の差分特徴を用いた破綻した話題遷移の検出

本研究では，発話間の差分ベクトルを用いて破綻した話題遷移を検出する．発話ベクトルの差分は，発話から発話で話題が遷移する方向を示しており，特定の方向や量の変化を起こした場合に破綻した話題遷移と紐づけられると考えられる．

破綻した話題遷移を検出する流れを図 1 に示す．はじめに，ユーザの発話とシステムの発話を word2vec を用いてベクトル表現に変換する．次に，これらの発話ベクトルより得られる特徴量を素性として抽出し，SVM を用いて破綻した話題遷移が生じているか識別する．SVM は，マージン最大化に基づいて識別面を選択して分類を行う教師あり機械学習手法である．汎化性能に優れていることやデータが少数でも識別が上手く機能することから，識別器として SVM を用いる．本研究では，SVM の実装に LIBSVM[1] を使用し，カーネルは RBF カーネルを使用する．分類対象のラベルは 3 種類であるため，one-versus-rest SVM を用いて多クラス分類を行う．

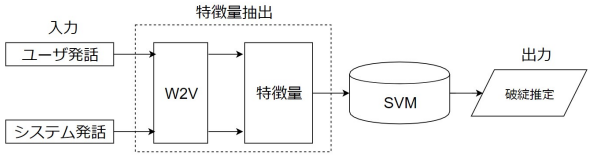

図 1: 破綻した話題遷移の自動検出モデル

3.1 word2vec を用いた発話文からの発話ベクトルの作成

発話ベクトルを作成する際に，word2vec[6] を用いて，単語を記号表現からベクトル表現に変換する．word2vec は，単語を固定長次元のベクトルとして特徴空間へ射影する．この手法では，分布仮説に基づき似た文脈上に存在する単語を近いベクトル空間上の点へと射影することができる．本研究では，日本語 Wikipedia より学習済みのモデル [11] を用いる．

発話ベクトルの作成にあたり，MeCab[4] を用いて形態素解析を行い，品詞情報に基づいて内容語 (名詞，動詞，形容詞) を抽出する．この際に，動詞や形容詞など用言の活用形はすべて終止形に戻す．次に，発話文より抽出した単語を word2vec を用いてベクトル表現に変換する．最後に，抽出された単語ベクトルの和によって，文ベクトルを以下の式で計算する (式 1)．文ベクトル (sentence_vector) を元に，ベクトルをノルム (ベクトルの大きさ) で割ることで計算される単位ベクトル (式 2) とノルム (式 3) を，発話文よりそれぞれ抽出し特徴量として用いる．各式の $\boldsymbol{w_i}$ は文中より抽出された内容語を word2vec によって変換したベクトル，n は内容語の総数を示している．

$$\text{sentence\_vector} = \sum_{i=1}^{n} \boldsymbol{w_i} \quad (1)$$

$$\text{unit\_vector} = \frac{1}{\text{norm}} \sum_{i=1}^{n} \boldsymbol{w_i} \quad (2)$$

$$\text{norm} = \sqrt{\sum_{i=1}^{n} \boldsymbol{w_i^2}} \quad (3)$$

3.2 発話ペアからの差分特徴量の抽出

本研究では，ユーザ・システム発話より発話ベクトル，ノルムを特徴量として抽出し，これらの特徴量の差分情報を素性として用いる．これらにより，ベクトルの遷移方向とその大きさが破綻の検出に有効であるかを調査する．差分ベクトルを検証する際に用いる素性の一覧を表 1 に示す．抽出された特徴量のみを使用する場合と，差分のみを使用する場合，これらを組み合わせた場合の 3 種類の素性を作成する．さらに，ベクトル，ノルム，ベクトルとノルムの 3 種類の組み合わせに対し，素性の作成を行うため，9 種類の素性が作成できる．各パターンに対し評価することで，どのような差分の情報が破綻した話題遷移の評価に対して効果的であるかを検証する．

表 1: 作成する素性一覧

素性名	用いる素性
Utr-vec	ユーザ・システム発話の発話ベクトル
Dif-vec	ユーザ・システム発話ベクトルの差分 (差分ベクトル)
Comb-vec	ユーザ・システム発話ベクトルと差分ベクトル
Utr-norm	ユーザ・システム発話のノルム
Dif-norm	ユーザ・システム発話ノルムの差分 (差分ノルム)
Comb-norm	ユーザシステム発話ノルムと差分ノルム
Utr-vec_norm	Utr-vec と Utr-norm
Dif-vec_norm	Dif-vec と Dif-norm
Comb-vec_norm	Utr-vec,Utr-norm,Dif-vec,Dif-norm

4 評価実験

3.2 項で提案した特徴量の中で，破綻した話題遷移の検出に対して効果的に機能するものがどのような特徴量であるか検証する．本研究では，話題遷移を含む対話破綻データを収集し，このデータを用いて破綻検出実験を行うことで，提案した特徴量の有用性について検証する．

4.1 実験データ

破綻ラベルが付与された学習データとして，DBDC2 で配布された学習データ (1,146 対話) と開発データ (150 対話) を用いる．しかし，このデータでは学習データに含まれるラベル数が NB に偏っており，このデータを用いて予備実験を行ったところ，識別器が上手く学習できない問題が明らかになった．そこでラベル数を均等にするために，学習データ中のラベル PB と B

のデータを複数回利用する (オーバーサンプリング). オーバーサンプリング前後の学習データ数の変化を表 2 に示す．ラベル NB のデータ数に近づけるため，ラベル PB のデータを 3 回，ラベル B のデータを 6 回，それぞれ再使用した．

本研究では，対話破綻の類型の中でも特に破綻した話題遷移の検出を目的としている．この目的に対する評価を行うためには，話題遷移を含んでいるユーザとシステムの発話ペアに対して，対話破綻のラベルが付与されたデータを用意する必要がある．しかし，話題遷移に着目して収集されたような雑談対話のコーパスは存在していない．そこで，DBDC で用意された評価データ (DBDC_eval) から，話題遷移を含んでいるユーザ・システム発話ペアを選別し，話題遷移が含まれたペアのみで構成されたデータを新たに作成した (TC_eval)．それぞれのデータ規模を表 3 に示す．また，TC_eval に含まれるデータの一例を表 4 に示す．

表 2: オーバーサンプリング前後の学習データ

	総データ数	ラベル NB	ラベル PB	ラベル B
Before	14,256	9,935	2,718	1,603
After	27,707	9,935	**8,154**	**9,618**

表 3: 評価データの規模

	総データ数	ラベル NB	ラベル PB	ラベル B
DBDC_eval	1,650	623	354	673
TC_eval	150	51	30	69

表 4: TC_eval のデータ例

ユーザ発話	システム発話	label
熱中症になったら大変だよ	予防が大切ですね	NB
ノリ軽いっすね	ノリは懐かしいですね	PB
今日の気分はどうですか	そうですね，あだ名はありますよ	B

4.2 評価尺度

DBDC の評価では，分布距離に基づいた尺度とラベル一致度に基づいた尺度が用いられている．本研究では，分類器の出力が分布ではなくラベルであることから，ラベル一致度に基づいた尺度を用いる．実際に使用する評価値は，正解率，適合率，再現率，F 値である．

4.3 実験結果

表 1 に示した 9 種類の素性を用いて SVM で対話破綻検出器を作成した．それぞれのモデルに対して，DBDC_eval，TC_eval を用いて評価を行った．表 5 は，9 種類の素性に対する正解率，適合率，再現率，F 値の評価結果を示している．適合率，再現率，F 値については，ラベル B の検出精度と，ラベル PB を B に含めた場合の検出精度をそれぞれ示している．また，表 5 の”BestScore”という項目は，DBDC3 で提案されたシステムの中で最もスコアが高いものを示している．

4.4 考察

DBDC_eval に対する評価結果について考察する．提案した素性の内，正解率および F 値 (B) では Dif-vec_norm，F 値 (B+PB) では Dif-vec が最も高い値を示している．DBDC3 の BestScore と比較した場合でも，正解率，F 値 (B) については，Dif-vec_norm が最も高い値を示している．しかし，F 値 (B+PB) では BestScore の値が最も高くなっている．この結果より，提案手法ではラベル PB をラベル NB と誤分類しているものが多いことが考えられる．

破綻した話題遷移について考察するため，TC_eval コーパスに対する評価結果について述べる．表 5 が示すように，正解率，F 値 (B) では Dif-vec_norm，F 値 (B+PB) では Dif-norm が最も良い結果となっている．また，Dif-vec_norm と Dif-norm の再現率 (B) を比較すると同値であることに対し，精度 (B) や F 値 (B) が向上している．これは，差分ノルム単体ではラベル PB や NB の発話を B に誤分類していたものが，差分特徴を組み合わせることで改善されたことを示唆している．一方で，再現率 (B+PB) や F 値 (B+PB) は差分特徴を組み合わせると低下している．これは，ベクトルやノルム自体の素性では PB と推定できていたものが，NB へ誤分類されている場合が多かった．

Dif-vec_norm の推定結果を混合行列として表現したものを表 6 に示す．ラベル NB とラベル B に対してはほぼ正しく推定できているが，ラベル PB が上手く推定できていないことが分かる．このことより，破綻していると断定できるようなユーザ・システム発話のペアに対しては適切な推定が出来るが，破綻しているか断定できないペアの推定が困難であることがわかる．

次に，TC_eval に対する評価結果の事例分析を行う．TC_eval の中から上手く推定できた結果を表 7 に示す．表のデータの中で，話題遷移を含むシステム発話として抽出したものは太字で記述している．一つ目の太字の発話では，ユーザの「焼きそばも好きです.」という発話に対して，焼きそばの麺の話題に展開するような「焼きそばは好きですか．麺が美味しいですね」という発話をシステムが行っている．これは，適切な話題遷移が行われている例であり，提案手法の検出器でも適切な発話であると正しく推定できている．二つ目の太字の発話では，ユーザがスポーツの話題に展開しようとしているのに対して，システムが話題を追従できず，以前会話していた麺の話題について発話しており，破綻した話題遷移が起こっている．これに対し，提案手法では破綻した話題遷移として正しく推定できている．

提案手法では，明らかに破綻した話題遷移が生じているシステム発話 (ラベル B が付与された発話) に対する検出は高いが，どちらとも取れないような発話 (ラベル PB が付与された発話) に対しては検出できておらず，破綻していないものとして検出してしまう．ラ

表 5: DBDC_eval と TC_eval に対する評価結果

素性名	DBDC_eval 正解率	適合率 (B)	再現率 (B)	F 値 (B)	適合率 (B+PB)	再現率 (B+PB)	F 値 (B+PB)	TC_eval 正解率	適合率 (B)	再現率 (B)	F 値 (B)	適合率 (B+PB)	再現率 (B+PB)	F 値 (B+PB)
Utr-vec	0.647	1.000	0.661	0.796	1.000	0.493	0.660	0.493	1.000	0.333	0.500	1.000	0.394	0.566
Dif-vec	0.364	0.31	0.242	0.272	0.816	0.789	**0.802**	0.460	0.5	0.478	0.488	0.808	0.771	0.789
Comb-vec	0.649	0.998	0.667	0.799	0.998	0.49	0.658	0.500	1.000	0.348	0.516	1.000	0.403	0.575
Utr-norm	0.458	1.000	0.196	0.328	1.000	0.234	0.379	0.526	1.000	0.406	0.577	1.000	0.367	0.537
Dif-norm	0.518	0.508	0.986	0.671	0.723	0.838	0.776	0.586	0.587	0.971	0.732	0.781	0.817	**0.798**
Comb-norm	0.384	1.000	0.016	0.032	1.000	0.023	0.045	0.347	1.000	0.014	0.029	1.000	0.037	0.715
Utr-vec_norm	0.672	1.000	0.722	0.839	1.000	0.492	0.659	0.526	1.000	0.406	0.577	1.000	0.367	0.537
Dif-vec_norm	**0.776**	0.997	0.979	**0.988**	0.997	0.59	0.742	**0.787**	1.000	0.971	**0.985**	1.000	0.614	0.761
Comb-vec_norm	0.694	1.000	0.776	0.874	1.000	0.475	0.644	0.553	1.000	0.464	0.634	1.000	0.321	0.486
BestScore	0.613			0.671			0.829							

表 6: TC_eval に対して Dif-vec_norm で評価した結果の混合行列

	prediction_NB	prediction_PB	prediction_B
answer_NB	**51**	0	0
answer_PB	30	**0**	0
answer_B	2	0	**67**

表 7: 上手く推定できている対話例

発話者	発話文	ans	pred
User	はい．どんなラーメンが好きですか？		
System	ラーメンは美味しいですね	PB	NB
User	焼きそばも好きです		
System	**焼きそばは好きですか．麺が美味しいですね**	**NB**	**NB**
User	そうですね．スポーツは好きですか？		
System	**麺類が美味しいですね**	**B**	**B**

ベル PB の推定は，人間でも一意に推定することが困難であることが考えられるため，対話破綻検出タスクの中でも難しい問題であることが考えられる．

5 おわりに

本研究では，雑談対話システムにおける破綻した話題遷移を検出する手法について述べた．具体的には，ユーザ・システムの発話ペアを入力とし，それぞれの発話文より word2vec を用いて発話ベクトル作成し，単位ベクトルやノルムの情報の差分特徴量を素性として SVM で学習することで，検出器を構築した．評価実験では，3 種類のラベルが付与された対話破綻検出コーパスを実験データとして用いた．また，話題遷移に着目するために，対話破綻検出コーパスから話題遷移を含むものを選別し，話題遷移を評価するためのデータを作成し，それぞれ評価を行った．実験の結果，発話ベクトルとノルムそれぞれの差分特徴量を素性としたモデルの正解率，F 値 (B) が DBDC3 の BestScore も含めて最も高くなった．一方で，提案手法がラベル PB をラベル NB と誤って推定してしまう問題もある．そのため，提案手法でラベル NB とラベル B に分類した後，ラベル NB と推定されたデータに対して，ラベル NB とラベル PB を分類するモデルを構築することにより，検出精度を向上させることが今後の課題として挙げられる．

謝辞

本研究の一部は JST CREST(課題番号: JPMJCR1513) および JST さきがけ（課題番号: JPMJPR165B）の支援を受けて行った．

参考文献

[1] Chih-Chung Chang and Chih-Jen Lin. Libsvm : A library for support vector machines. In *ACM Transactions on Intelligent Systems and Technology*, Vol. 2, 2011.

[2] Ryuichiro Higashinaka, et al. Fatal or not? finding errors that lead to dialogue breakdowns in chat-oriented dialogue systems. In *SIGDIAL*, pp. 11–16, 2015.

[3] Ryuichiro Higashinaka, et al. Overview of dialogue breakdown detection challenge 3. In *Diaog System Technology Challenges 6*, 2017.

[4] Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. Applying conditional random fields to japanese morphological analysis. In *EMNLP*, pp. 230–237, 2004.

[5] Bilyana Martinovsky and David Traum. The error is the clue: Task description, datasets, and evaluation metrics. In *ISCA Workshop on Error Handring in Spoken Dialogue System*, pp. 11–16, 2003.

[6] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013.

[7] Hiroaki Sugiyama. Dialogue breakdown detection based on estimating appropriateness of topic transition. In *Diaog System Technology Challenges 6*, 2017.

[8] Junya Takayama, Eriko Nomoto, and Yuki Arase. Dialogue breakdown detection considering annotation biases. In *Diaog System Technology Challenges 6*, 2017.

[9] 塚原祐史, 内海慶. 対話行為と話題推定によるラベル伝搬を利用した雑談生成方法の改良. 第 30 回人工知能学会年次大会, 2016.

[10] 東中竜一郎ほか. 対話破綻検出チャレンジ 2. In *SIG-SLUD*, pp. 81–84, 2016.

[11] 鈴木正敏ほか. Wikipedia 記事に対する拡張固有表現ラベルの多重付与. 言語処理学会第 22 回年次大会, 2016.