Vol.2020-ICS-199 No.8
2020/3/24

BERTと文章の意味的特徴量を用いた誤差予測モデルによる対話破綻検出

松本 丈樹[1]　藤田 桂英[2]

概要：近年，自然言語処理分野の発展とともに，人と機械が対話をするような対話システムが多く開発されている．対話システムの目的は様々であるが，特に会話そのものを目的とした対話システムでは，対話の継続が重要な課題の一つである．しかし，多くの対話システムにおいて，ユーザはスムーズに対話を継続することが困難な状況である，対話破綻に陥る．このような対話破綻を検出するようなタスクにおいて，既存手法であるBERTを使った手法は，BERTに他の特徴量として文章の長さやシステムの発話ターン数，文章ベクトル，類似度，発話の種類などを加えることで精度の向上を測ったものである．ベースにBERTを用いたこともあり，比較的高い精度を出すことに成功している．しかし，BERTのfine-tuningやその他の特徴量の抽出方法など，詳細な記載が無いため再現性が低い研究となっている．そこで本研究では，BERTと他の特徴量を用いた対話破綻検出手法について再現性をもって実装し，他の手法との比較，新たな特徴量の調査や評価を行うことを目的とした．提案手法として，fine-tuning後のBERT出力の誤差を予測するモデルを実装した．また，新たな特徴量としてJumanの形態素解析の際に得られるドメイン情報，意味カテゴリ情報を考慮した．評価実験の結果，誤差を予測するモデルは先行研究と同程度の性能を示すことに成功した．また，ドメイン情報，カテゴリ情報の付与により，BERTの出力が改善することが確認された．特徴量別にみると，文章ベクトルの類似度を付加した場合の精度向上が最も大きいことから，少ない次元数でも適切な特徴量を与えてやればBERTの誤差予測に効果的に働くことがわかった．

キーワード：対話破綻検出，BERT，誤差予測

1. はじめに

対話システムは2種類に大別される．タスク指向型対話システムと，非タスク型対話システムである．タスク指向型対話システムとは，何かの目的のために人 (ユーザ) と機械 (システム) が対話を行うというもので，多くの場合，特定のドメインの特定の問題を解決するための手段として対話が用いられる形となる．一方で非タスク指向型対話システムでは，主に対話をすること自体が目的となる．特に，特定のドメインに縛られず，特定の問題解決を目的としない対話，すなわち雑談を継続して行うことが目的になることが多い．雑談ではドメインが限定されていないため，システムがユーザの発話を想定し辛いという特徴がある．そのためタスク指向型対話システムに比べると，一般的にシステムを開発するのが難しくなる．近年の自然言語処理の発展により，ニューラルネットワークを駆使したシステムが多く開発されているが，多くのシステムにおいては十分に雑談を行うことができているとは言い難い状況にある．以下に示すのがその一例である．

対話破綻の一例 (データセット (2_dev) から抜粋)

U1: あなたは何人家族ですか？
S1: 私は、4人家族ですね。
U2: 兄弟や姉妹はいますか？
S2: 年越しそばを食べます

ユーザの問いかけ (U2) に対するシステムの応答 (S2) が文脈からそれたものになってしまっている．ユーザの発話に即していなかったり，自身の以前の発言に即していないような発話をシステムが行ってしまうと，ユーザは対話を継続することが難しいと感じてしまう．このような状況を対話破綻と呼ぶ．対話破綻が起きてしまうと，ユーザは対話を続けることにストレスを感じてしまうため，対話が終了してしまう可能性が高くなる．これは対話の継続という非タスク指向型対話システムの主目的の一つに反する状況

[1] 東京農工大学工学部
Faculty of Engineering, Tokyo University of Agriculture and Technology
[2] 東京農工大学大学院工学研究院
Institute of Engineering, Tokyo University of Agriculture and Technology

であり，対話破綻を起こさないようにすることが対話システムにおいて重要な課題である．

対話破綻を起こさないようにするための取り組みの第一歩として，既存の対話のログの中から対話破綻を検出するというタスクがあり，これを対話破綻検出という．対話破綻検出を正しく行うことができれば，対話破綻に陥るのを回避したり，対話破綻から回復したりするのに有用である．

対話破綻検出に注目した取り組みとして，対話破綻検出チャレンジ [1][2][3][4] という競技会が 2015 年から開催されている．これはシステム発話とそれまでの対話ログが与えられたときに，システム発話の対話破綻の度合いを予測してその精度を競う大会である．ユーザとシステムが雑談を行っている対話ログ内のすべてのシステム発話に対して，あらかじめ対話破綻の度合いがアノテーションされているデータが与えられるので，未知の対話ログに対してその対話破綻度合いを予測するモデルを作成し，その予測精度を競う．

この対話破綻検出チャレンジにおいて，Sugiyama によって BERT とその他の特徴量を用いた手法 [5] が提案された．BERT[6] は 2018 年に Google によって開発された汎用学習モデルで，様々なタスクにおいて SOTA を更新した高性能な言語モデルである．実際に Sugiyama の手法は高い精度を出すことには成功している．しかし，詳しい実験設定などが記載されていないため再現性が低いことや，実験結果の他の手法と比較されていないことなどの問題点がある．そこで本研究では，Sugiyama の手法に基づいて BERT と他の特徴量を用いた誤差予測モデルを実現した．また，新たな特徴量を追加することで対話破綻検出精度の向上を目指した．

本研究における目的は以下のとおりである．

- BERT と他の特徴量を用いた手法を再現性を持って実装する．
- BERT と他の特徴量を用いた手法を既存の他の手法と比較する．
- BERT と他の特徴量を用いた手法で用いる各特徴量について，新たな特徴量を導入した上で各特徴量について評価を行う．

本論文の構成について述べる．まず，本研究に関連性の高い研究として，主に対話破綻検出チャレンジと対話破綻検出の既存手法について述べる．次に，Sugiyama の手法 [5] を基にした提案手法である，BERT fine-tuning モデルと誤差予測モデルについて，詳細を述べる．その後，評価実験を行い各特徴量の評価，他の手法との比較を行う．最後に，本研究のまとめと，今後の課題について述べる．

2. 関連研究

2.1 対話破綻検出チャレンジ

対話破綻検出チャレンジ (DBDC: Dialogue Breakdown Detection Challenge)[1][2][3][4] という対話破綻検出の精度を競うコンペティションが開催されている．参加者には開発用対話ログと，評価用対話ログが与えられる．対話ログはユーザとシステムが交互に発話したものから成り，すべてのシステムの発話には対話破綻度合いがアノテーションされている．(図 1) 対話破綻度合いは以下の 3 つのラベルが付与されている．

	発話	○	△	×
システム	こんにちは。ちょうど話したいと思ってました			
ユーザ	こんにちは。私もです。			
システム	お気に入りの番組はありますか？	23	4	3
ユーザ	いくつかありますよ。			
システム	最近読んで面白かった本とか、おすすめ教えてもらえます？	13	14	3
ユーザ	「仮面の商人」という本が面白かったですよ。			
システム	商人は頭がよくないじゃないですか？	11	11	8
ユーザ	そうですね。頭の回転が速そうですね。			
システム	商人は地位が低いですね	10	11	9
ユーザ	そうですか？			
システム	そうだよ	26	4	0
ユーザ	どうしてそう思いますか？			
システム	商人の街ライフは快適だぜ	5	7	18
ユーザ	楽しそうですね。			
システム	めっちゃ楽しい	24	5	1
ユーザ	あなたは商人なのですか？			
システム	男です	3	6	21
ユーザ	そうですか。			
システム	そうだよ	4	6	20
ユーザ	お年はおいくつですか？			
システム	男は勝手です	3	1	26

図 1 DBDC 対話ログとアノテーションの例 (図は [7] より引用)

対話破綻ラベル

- ○ 破綻ではない．以降の対話をスムーズに継続できる．
- △ 破綻ではないが，違和感を感じる．以降の対話をスムーズに継続することが困難．
- × 破綻である．以降の対話を継続することが困難．

それぞれのシステム発話に対して 30 人のアノテータによってラベル付けが行われている．一般的な分類問題と異なり，各ラベルは対話破綻の度合いを表していることや，アノテータによって判断が異なる主観的なアノテーションであるため，単純な分類問題であるとは考えにくい．そこで DBDC では，2 種類の評価方法が用いられる．1 つ目は，単純な分類問題として各システム発話に対して単一の対話破綻ラベルを予測させるというものである．各システム発話の正解ラベルをアノテーション結果から多数決で決定し，これを正解データとして学習・評価を行う．一般的な分類問題と同様であるため，評価尺度は一致率，適合率，再現率，F 値などを用いることができる．2 つ目は，各システム発話における，付与されたラベルの割合を予測するというものである．各システム発話の正解ラベルをアノテーションで付与された対話破綻ラベルの割合とすることで，これを正解データとして学習・評価を行うことができる．これは予測モデルに 3 つのクラスに分類される確率を確率分布として予測させるタスクと見なすことができる．以上から，評価尺度は正解の分布と予測した確率分布との間の

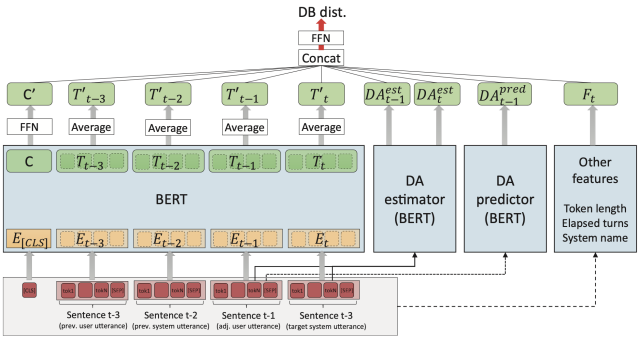

図 2 BERT + 他の特徴量モデルの概形 (図は，論文 [5] より引用)

距離を用いることができる．具体的には，JS-Divergence，Mean Squared Error を用いる．

2.1.1 対話破綻検出手法

JongHo ら [8] は BiLSTM と Self-Attention の組合わせモデルを用いた手法を用いて，対話の流れを考慮した対話破綻検出手法を提案した．大小 2 つの注意機構を取り入れることで，レアなケースの対話破綻をより効果的に検出することを目指したものである．結果的に，対話の流れを考慮することは深いニューラルネットワークを用いた対話破綻検出において重要であることが示された．

Mariya ら [9] は LSTM の組み合わせ方と分散表現の種類の最適な組合わせを検証した．モデルとして単層の LSTM，多層の LSTM，Bi-LSTM を，分散表現として Google News データから作成した Word2Vec，Twitter データから作成した GloVe Twitter，巨大コーパスから作成した GloVe Common Crawl を比較した．結果的に，単層 LSTM と Glove Common Crawl の組合わせが最も最適であると結論づけた．

桑原ら [10] は Universal Transformer[11] で学習された対話モデルを用いた対話破綻検出手法を提案した．RNN を用いた手法との比較で精度向上が確認され，Self-Attention が文構造や意味構造を学習するのに有用であることが確認された．しかし適合率が RNN を用いた手法を下回る結果となり，適合率を向上させるような特徴量を検討する必要があると結論づけた．

2.1.2 BERT + 他の特徴量を用いた対話破綻検出モデル

DBDC4 において Sugiyama が提案した手法 [5] について説明する．BERT の出力に，入力の文章から得られる他の特徴量を組み合わせた対話破綻検出手法である．モデルの概形は以下の図 2 のようになっている．

特殊トークンである [CLS] に対応する BERT 出力部分にニューラルネット層を追加し，対話破綻ラベルの割合を正解ラベルとして fine-tuning した BERT を用意する．入力として用いる文章は，対話破綻検出を行うシステム発話とその直前の 3 つの発話 (1 つ前のユーザ発話，1 つ前のシステム発話，2 つ前のユーザ発話) である．BERT の出力結果に対して入力文から得られる他の特徴量を Concat したものをニューラルネット層に入力することで，最終的な出力を得るモデルとなる．

他の特徴量として用いるのは，文章の長さ，システム発話の対話におけるターン数，対話システムの名称，文章ベクトルとその類似度，Dialogue Act である．文章ベクトルは，BERT の最終出力のうち各トークンに対応するベクトルを，各発話について平均をとったものとしている．また，Dialogue Act は発話の種類を示すもので，BERT を用いた別のモデルで推測されるものを用いる．

このモデルについて，DBDC のデータセットを用いて評価実験が行われた．BERT だけのモデルと比較して，文章ベクトルとその類似度を付加したモデルでは Accuracy で 0.0012 ポイント，すべての特徴量を付加したモデルでは Mean Squared Error で 1.5%の改善が確認された．

このことから，対話破綻検出においては，BERT に対して他の特徴量を付加することは効果的であることが伺える．ただしパラメータ設定やニューラルネットワーク層，特徴量の抽出方法についての詳細な記述がないために再現性の低いものとなっている．

3. 対話破綻検出を活用した研究

稲葉ら [12] は対話破綻検出を対話システムに適用することで，対話システムの応答性能を向上させる取り組みを行った．対話破綻ラベルの分類結果を用いる手法，破綻でない確率を用いる手法，破綻確率分布と応答スコアを線形回帰の入力として用いる手法を提案した．結果として，後者 2 つの手法についてはシステムの応答性能の向上が確認された．また，対話破綻検出を適用した対話システムの特徴として，対話の自然さは向上するが，対話の多様性が低下することによりユーザが対話を楽しいと感じにくくなってしまうことが確認された．ただし総合的にみると対話破綻検出を適用した方が良い結果を出したため，今後は応答の内容に注目した対話破綻検出の適用手法を検討していく必要があると結論づけた．

4. BERT fine-tuning と 誤差予測モデル

Sugiyama の手法 [5] を基にした，BERT の予測を他の特徴量で補正するようなモデルである．モデルの概観は図 3 のようになる．

このモデルは 2 つのニューラルネットモデルの組合わせからなる．1 つ目は，日本語で事前学習済みの BERT を対話破綻検出タスクに対して fine-tuning したモデル．2 つ目は，fine-tuning した BERT の出力と正解ラベルとの誤差を予測するモデルである．

4.1 BERT fine-tuning モデル

1 つ目の BERT を fine-tuning するモデルについて説明

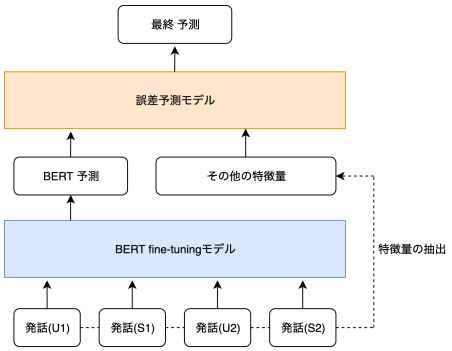

図 3　BERT fine-tuning と 誤差予測モデルの概形

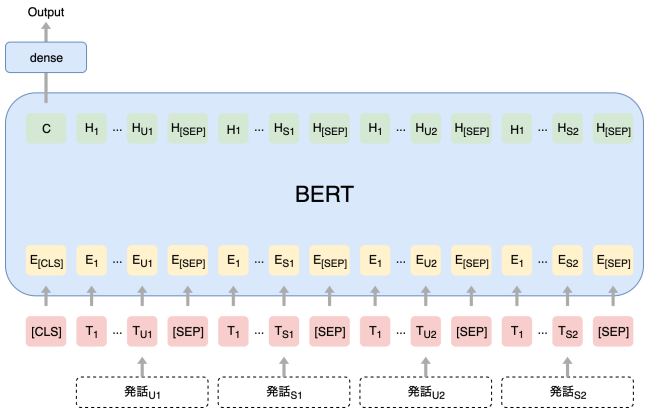

図 4　BERT fine-tuning モデル

する．大規模な日本語のテキストデータで事前学習を済ませた BERT に対して，fine-tuning(転移学習) を行って対話破綻ラベルの予測を行うことができるようにしたモデルである．モデルの概形は図 4 のようになる．

BERT のエンコーダの最終層の出力のうち，BERT 特殊トークンである [CLS] トークンに対応する出力の上層に，1 層の全結合層を付け足したモデルになる．対話破綻検出をしたいシステム発話とその直前の 3 つの発話 (1 つ前のユーザ発話，1 つ前のシステム発話，2 つ前のユーザ発話) の計 4 発話を BERT に入力し，各対話破綻ラベルの割合を出力する．

BERT の事前学習モデルは，京都大学の黒橋・河原研究室が公開している BERT 日本語 Pretrained モデル [13] を用いた．モデル構造は $BERT_{BASE}$ と同様に 12 層，12 ヘッドで，埋め込みベクトルのサイズは 768 次元である．日本語 Wikipedia を用いて事前学習が行われた．分かち書きの処理には Juman++と BPE を用いた subword 化が使われ，BERT ID に変換できる語彙数は 32,000 語である．

4.2　誤差予測モデル

2 つ目の BERT の誤差を予測するモデルについて説明する．上記の fine-tuning 済みの BERT の予測結果と入力文から抽出した特徴量を入力として受け取り，対話破綻ラベルの割合を出力するモデルである．モデルの概形は図 5 のようになる．fine-tuning 済みの BERT の出力と文章から得られた特徴量を入力として，これらを Concat して全結合層にいれる．出力は BERT 出力と正解データとの誤差であるから，これを BERT 出力に加算して予測結果とする．

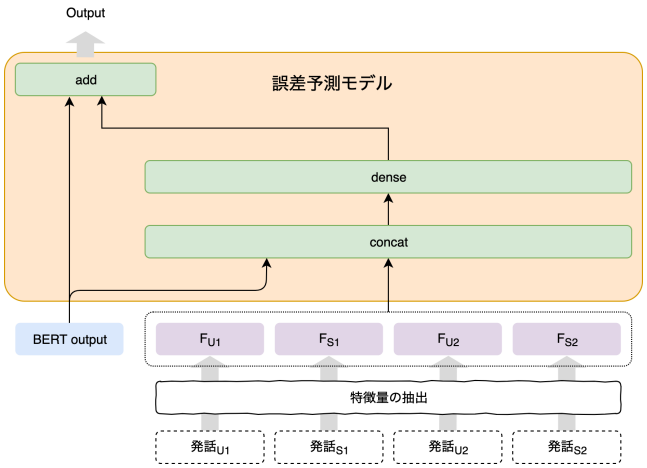

図 5　誤差予測モデルの概形

誤差予測に用いた特徴量とその抽出方法について，以下に述べる．

発話の長さ

発話に含まれる文章をすべて Juman++によって分かち書きした際の分割数を，発話の長さと定義する．入力に用いる発話は 4 つなので，1 つの入力に対して 4 つの発話の長さが得られる．これを Concat して 4 次元の特徴量として用いる．

文章ベクトル

各発話を固定長のベクトルに変換したものを文章ベクトルとする．Word2Vec と BertVec による文章ベクトルの獲得方法について述べる．

Word2Vec での埋め込みには，日本語大規模 SNS+Web コーパスによる単語分散表現モデル [14] で公開されている学習済みのものを用いた．各発話を Juman++で分かち書きにし，すべてを Word2Vec で 200 次元の分散表現に埋め込む．発話に含まれる分散表現の平均を文章ベクトルとして得る．4 発話から得られた文章ベクトルを Concat し，200 × 4 次元次元の文章ベクトルを得る．

BertVec での埋め込みには，BERT fine-tuning モデルを使用する．モデルの最終隠れ層の出力 (トークン数× 768 次元) を各トークンの分散表現と考え，これを発話ごとに平均することで各発話に対して 768 次元の文章ベクトルを得る．4 発話から得られた文章ベクトルを Concat し，4 × 768 次元の文章ベクトルを得る．

文章類似度

文章類似度は，文章ベクトル間のコサイン類似度を計算することで得られる．BertVec によって求めた 4 つの文章ベクトルのすべての組合わせについてコサイン類似度を計算することで 6 つの値を得る．これを Concat して 6 次元

の特徴量として用いる．

ドメイン情報

各発話を Juman++で形態素解析を行うと，分かち書きした各トークンにドメイン情報が与えられることがある．ドメイン情報は「文化・芸術」,「レクリエーション」,「スポーツ」,「健康・医学」,「家庭・暮らし」,「料理・食事」,「交通」,「教育・学習」,「科学・技術」,「ビジネス」,「メディア」,「政治」の 12 種からなる [15]．このドメイン情報を用い，発話単位でドメインベクトルを作成した．ドメインベクトルは 12 種類のドメイン情報に「ドメイン情報無し」の 1 種類を加えた 13 種類を各要素にもつベクトルで表され，各ドメインの発話内での出現回数をカウントして各要素の値とした．発話内にドメイン情報を持つトークンが 1 つも存在しないときは,「ドメイン情報無し」をカウントする．最後にドメインベクトルを要素の和が 1 になるように正規化して，各発話に対するドメインベクトルを得た．

意味カテゴリ情報

各発話を Juman++で形態素解析を行うと，分かち書きした各トークンに意味カテゴリ情報が与えられることがある．意味カテゴリ情報は「人」,「組織・団体」,「動物」,「植物」,「動物-部位」,「植物-部位」,「人工物-食べ物」,「人工物-衣類」,「人工物-乗り物」,「人工物-金銭」,「人工物-その他」,「自然物」,「場所-施設」,「場所-施設部位」,「場所-自然」,「場所-機能」,「場所-その他」,「抽象物」,「形・模様」,「色」,「数量」,「時間」の 22 種からなる [15]．この意味カテゴリ情報を元に，ドメインベクトルと同様の方法を用いてカテゴリベクトルを作成した．カテゴリベクトルは 22 種類の意味カテゴリ情報に「意味カテゴリ情報無し」の 1 種類を加えた 23 種類を各要素にもつベクトルで表され，各意味カテゴリの，発話内での出現回数をカウントして各要素の値とした．発話内に意味カテゴリ情報をもつトークンが 1 つも存在しないときは,「意味カテゴリ情報無し」をカウントする．最後にカテゴリベクトルを要素の総和が 1 になるように正規化して，各発話に対するカテゴリベクトルを得た．

Juman++から得られる意味カテゴリ情報，ドメイン情報は，各トークンの意味を表したものである．したがってこれらの情報を発話単位でベクトル化することにより，発話単位での意味を表す特徴量として扱うことができるのではないかと考えた．文章の意味を表す特徴量を用いるという点では，文章ベクトルを特徴量として用いることと似ているが，文章ベクトルが言語モデルが与えた高次元のベクトルであるのに対し，ドメイン情報，意味カテゴリ情報を用いたベクトルは，辞書ベースの低次元ベクトルであるという違いがある．

5. 評価実験

5.1 データセットの作成

過去の DBDC で用いられたデータセットのうち, DBDC2 開発用，DBDC2 評価用，DBDC3 評価用のもの[*1]を用いて，実験に使用するデータセットを作成した．データセットに含まれる各対話は，ユーザ (人) とシステムが交互に発話する形式で構成され，システム発話から始まりそれぞれ 10 回発話したものが 1 つの対話となっている．また，すべてのシステム発話には 30 人アノテータによって対話破綻ラベルが付けられている．これらのデータセットから，タスクに合わせて入力データと出力データに分割する．今回使用するモデルでは，各システム発話と直前の 3 つの発話の計 4 発話が入力データ，対話破綻検出対象となるシステム発話の対話破綻ラベルの割合が出力データとなる．DBDC データセットに含まれる各対話から，すべてのシステム発話に対応する入力データ，出力データを計算し，DBDC2 の開発用のデータセットから 1500 個，DBDC2 の評価用データセットから 1500 個，DBDC3 の評価用データセットから 1500 個の入出力データを得た．

5.2 実験の目的

BERT fine-tuning モデルと誤差予測モデルの性能を検証するために，以下の実験を行った．

- BERT fine-tuning モデルを学習させ，既存手法と比較し評価する．
- 誤差予測モデルによる BERT fine-tuning モデルの精度改善の度合いを，特徴量ごとに比較することで，効果的に働く特徴量について評価する．

5.3 BERT fine-tuning モデルの評価

第 4 章で述べた BERT fine-tuning モデルについて，学習を行った．モデルの学習に DBDC2 開発用と DBDC2 評価用，評価に DBDC3 評価用のデータセットを用いた．最適化アルゴリズムとして AdamW[16] を，学習率 $2e-5$ で用いた．入力の最大長 512，ミニバッチサイズ 12 で GPU(Quadro GP100)1 台を用いて fine-tuning を行った．

図 6 は実験の過程を示している．エポック 2 までは順調に学習を進め，Loss が低下，Accuracy が向上していることがわかる．一方，エポック 2 以降は過学習が始まり，Loss が増加，Accuracy が低下している．過学習を行ったモデルを誤差モデルの元データとして用いるのは望ましくないと考えられる．したがってエポック 2 を学習後のモデルを BERT fine-tuning モデルとして採用することにした．

表 1 は，評価に DBDC3 評価用データセットを用いている既存手法の論文に記載されている結果との比較を示してい

[*1] https://dbd-challenge.github.io/dbdc3/data/

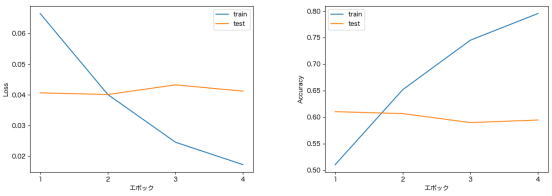

図 6 fine-tuning 時の，Loss と Accuracy の変化の様子

表 1 BERT fine-tuning モデルと既存手法との比較

	Loss(MSE)	Accuracy
Sugiyama (original BERT)	0.0401	0.6018
OUARS (LSTM + CNN)	0.0465	0.5669
PLECO (MemNN)	0.0517	0.5386
BERT fine-tuning モデル	0.0430	0.6033

る．OUARS(LSTM + CNN)，PLECO(MemNN) などの BERT を用いない既存手法 [3] と比較すると，fine-tuning モデルは低い Loss を記録していることがわかる．Sugiyama の論文に記載のあった fine-tuning の結果の Loss(0.0401) と比較すると今回の fine-tuning モデルでは Loss(0.0430) と 7.2%増加してしまっている．原因として考えられるのは，Sugiyama の手法ではより多くの学習データを用いていることである．今回の BERT fine-tuning モデルで用いた学習データは，DBDC2 の開発用，評価用データセットの 2 つであるが，Sugiyama の手法ではこれらに加え，DBDC1 の開発用，評価用データセットも用いている．また，Sugiyama の論文にはハイパーパラメータなどの詳細な記載がなかったため，今回使用したものとは異なる設定で fine-tuning を行っていたと考えられる．

5.4 誤差予測モデルにおける精度の改善

BERT fine-tuning モデルの予測結果と正解データとの誤差を予測する誤差予測モデルについて，fine-tuning モデルからどの程度精度が向上するのかについて実験を行った．前節で述べた BERT fine-tuning モデルのうち最も精度が高かった fine-tuning モデルの予測結果を誤差予測モデルの入力として用いた．目的関数は，正解分布と予測分布の MSE(Mean Squared Error) を用いることで統一した．各特徴量を付与したモデルを学習させた際の結果は表 2 のようになった．

表 2 は，各特徴量を付与した誤差モデルを学習させた際に，最もよい性能を示したエポックでの Loss(損失) と Accuracy(精度) を表している．Loss は目的関数である MSE を用いて計算した．Accuracy は各データの正解データのうち最も高い割合のクラスを正解クラスとしたときの正解率を用いて計算した．Loss の値を減少させた特徴量は，発話の長さ (-0.80%)，文章類似度 (-3.32%)，意味カテゴリ情報 (-0.86%)，ドメイン情報 (-1.24%)，Loss の値を増加させた特徴量は，文章ベクトル (Word2vec:+13.27%，

表 2 誤差予測モデルの特徴量別の Loss(MSE) と Accuracy

	Loss(MSE)	Accuracy
fine-tuning モデル	0.04307	0.6033
発話の長さ	0.04272 (-0.80%)	0.5960 (-0.73%)
文章類似度	0.04164 (-3.32%)	0.6047 (+0.14%)
文章ベクトル		
-Word2Vec	0.04879 (+13.27%)	0.5980 (-3.53%)
-BertVec	0.04399 (+2.14%)	0.6147 (+1.14%)
ドメイン情報	0.04254 (-1.24%)	0.5980 (-0.53%)
意味カテゴリ情報	0.04270 (-0.86%)	0.5980 (-0.53%)

BertVec:+2.14%)，精度を向上させた特徴量は，文章類似度 (+0.14%)，文章ベクトル (BertVec:+1.14%)，精度を低下させた特徴量は，発話の長さ (-0.73%)，文章ベクトル (Word2Vec:-3.53%)，意味カテゴリ情報 (-0.53%)，ドメイン情報 (-0.53%) であった．

Loss と Accuracy の関係に注目すると，Loss を低下させることと，Accuracy を向上させることは必ずしも同じではないことがわかる．これは対話破綻検出が単純な分類タスクではないことが原因である．単純な分類タスクの場合，正解ラベルは one-hot で表されるため，確率分布で見ると正解のクラスのみが 1 で，他のクラスが 0 で表されている．しかし対話破綻検出のように正解ラベルが各クラスの確率分布で表される場合，正解クラスと他のクラスの確率にあまり差がないようなことがある．例えば 3 クラスで (0.35, 0.33, 0.32) が正解ラベルとなるデータがあったとき，予測 (1.0, 0.0, 0.0) では Loss は大きいが予測クラスは正しく，予測 (0.33, 0.32, 0.35) では Loss は小さいが予測クラスは誤りとなる．このように対話破綻検出では，Loss を低下させると予測クラスが変わるということが起きてしまう．

以下に，各特徴量に関して，考察を述べていく．

文章の長さ・文章類似度

表 2 から，文章類似度を用いた誤差モデルが Loss を 3.32%減少させていることがわかる．これは今回用いた特徴量の中で最もよい結果である．特徴量の次元数は 6 次元と小さい特徴量であるにもかかわらず，同じ低次元の特徴量である発話の長さの Loss(-0.08%) と比べてもよい結果を出している．よって誤差モデルは，低次元の特徴量でも適切な特徴量を与えてやれば効果的に Loss を減少させることができるといえる．

文章ベクトル

表 2 から，文章ベクトルを用いた誤差予測モデルでは，Word2Vec，BertVec どちらを用いた場合でも Loss が増加していることがわかる．Loss を増加させた特徴量は文章ベクトルのみである．文章ベクトルと他の特徴量の大きな違いは次元数である．Word2Vec を用いた場合は 200 × 4=800 次元，BertVec を用いた場合は 768 × 4=3072 次元の特徴量となる．これらのことから，誤差モデルにおいては高次元ベクトルはノイズとしての働きが大きいと考えら

表 3　各データセットにおける，ドメイン情報を含むトークンの割合 (%) とドメイン情報を含む発話の割合 (%)

	2_dev	2_eval	3_eval
トークンの割合	9.37	9.75	9.39
発話の割合	56.00	56.57	57.65

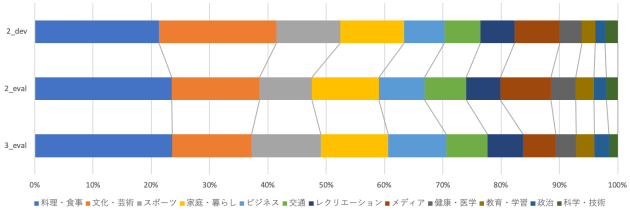

図 7　ドメイン情報の内訳 (上から，DBDC2 開発用，評価用，DBDC3 評価用データセット)

れる．誤差モデルは全結合層で構成されるシンプルなモデルであるから，高次元の特徴量から効果的に学習することが難しい．また，誤差を発生させる元となるモデルである BERT fine-tuning モデルが高精度であるため，文脈に依存しない Word2Vec を用いた際には Loss が大きく増加する結果になった．

ドメイン情報

表 3 は，各データセットにおける，ドメイン情報を含む分かち書きトークンの割合と，ドメイン情報を含むトークンを少なくとも 1 つ含む発話の割合を示している．これをみると，全トークンに対して 1 割程度しかドメイン情報をもつトークンがない．発話単位でみると，約半数が何らかのドメイン情報をもつトークンを含んでいる．また，これらの割合はデータセットに依らないことがわかる．これらのことから，ドメイン情報を学習に使用する際にはデータセットによる特徴量の有無の偏りは問題にならないと考えられる．

表 2 の結果から，ドメイン情報を用いた誤差モデルは Loss を 1.24%減少させていることがわかる．これは文章類似度を用いた誤差モデルの Loss(-3.32%) に次いでよい結果であることから，BERT の誤差を予測するのに効果的に働く特徴量である．また，約半数にしか含まれていないような特徴量を用いても，効果的に働く場合があることがわかる．

図 7 は，各データセットにおける，ドメイン情報の内訳を示している．ドメイン情報の内訳に，学習データと評価データで大きなばらつきがある場合，データ数の少ない今回のようなケースでは特徴量として適切でないだろう．学習データに多く含まれるようなドメインに強いモデルになってしまう可能性があるからだ．しかし図をみてみると，順位こそ違うが同じような分布をしているといえるだろう．これはドメイン情報が Loss を減少させていることからも妥当である．

表 4　各データセットにおける，意味カテゴリ情報を含むトークンの割合 (%) と意味カテゴリ情報を含む発話の割合 (%)

	2_dev	2_eval	3_eval
トークンの割合	19.97	19.51	18.96
発話の割合	82.86	81.02	82.67

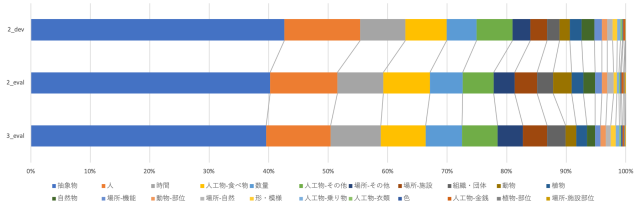

図 8　意味カテゴリ情報の内訳 (上から，DBDC2 開発用，評価用，DBDC3 評価用データセット)

意味カテゴリ情報

表 4 は，各データセットにおける，意味カテゴリ情報を含む分かち書きトークンの割合と，意味カテゴリ情報を含むトークンを少なくとも 1 つ含む発話の割合を表している．これをみると，トークン単位では約 20%が意味カテゴリ情報をもっており，80%以上の発話が何らかの意味カテゴリ情報をもつトークンを含んでいる．ドメイン情報と同様にデータセットによる特徴量の有無の割合は気にしなくてよいと考えられる．

表 2 の結果をみると，意味カテゴリ情報は Loss を 0.86%減少させていることがわかる．これはドメイン情報の Loss(-1.24%) よりも変化量が少ない．特徴量を含む発話はドメイン情報より意味カテゴリ情報の方が多いが，変化量はドメイン情報の方が多いという結果になった．原因として考えられるのは，意味カテゴリ情報の内訳である．図 8 は各データセットに含まれるカテゴリ情報の内訳を示している．これをみると，意味カテゴリ「抽象物」がどのデータセットでも 40%近くを占めていることがわかる．「抽象物」は意味カテゴリの中でも抽象度の高いカテゴリであり，実際にはより多くの意味カテゴリを含んでいる．このように抽象度の高いカテゴリがデータセットに多く含まれていたため，ドメイン情報に比べて精度の向上が低かったと考えられる．

6. まとめと今後の課題

本論文では，人 (ユーザ) と機械 (システム) が対話する対話システムにおける対話破綻検出を行う手法として，BERT fine-tuning モデルとその誤差をその他の特徴量で予測するモデルを組み合わせたもの提案した．また誤差予測モデルで用いる特徴量として，新たに Juman++のドメイン情報，意味カテゴリ情報を考慮した．BERT fine-tunig モデルでは，既存研究の結果に及ば無い結果となってしまったが，これは使用したデータセットやハイパーパラメータの差によるものと考えられる．誤差予測モデルの評価実験に

より，誤差予測モデルにおいては適切な特徴量を与えてやれば，次元数の低い特徴量であっても効果的にはたらくことがわかった．また，ドメイン情報，意味カテゴリ情報は誤差予測モデルに効果的にはたらくことがわかった．

BERT を fine-tuning するだけでも，かなりの高精度なモデルができることが実験により確認できた．しかし今回用いた事前学習モデルは，Wikipedia のデータから学習しているため，書き言葉に強い言語モデルである．一方で今回のタスクである対話破綻検出は，元となるデータが対話文である．対話文においては，話し言葉がとても重要な役割を果たしていると考えられる．Wikipedia のデータを学習するだけでは，対話文特有の話し言葉に強い言語モデルを作成することは困難である．以上から，対話文のデータを元に事前学習を行ったような BERT モデルを作成することが，対話文に関する研究全体を発展させる上で重要な役割を果たすと考えられる．

参考文献

[1] 東中竜一郎，船越孝太郎，小林優佳，稲葉通将ほか：対話破綻検出チャレンジ，*SIG-SLUD*, Vol. 5, No. 02, pp. 27–32 (2015).
[2] 東中竜一郎，船越孝太郎，稲葉通将，荒瀬由紀，角森唯子ほか：対話破綻検出チャレンジ 2，*SIG-SLUD*, Vol. 5, No. 02, pp. 64–69 (2016).
[3] R. Higashinaka, K. Funakoshi, M. I. Y. T. T. T. and Kaji, N.: “Overview of Dialogue Breakdown Detection Challenge 3，*in Proceedings of Dialogue System Technology Challenge 6 (DSTC6) Workshop* (2017).
[4] Ryuichiro Higashinaka, Luis F. D’Haro, B. A. S. R. E. B. K. F. M. I. Y. T. T. T. J. S. : Overview of the Dialogue Breakdown Detection Challenge 4, *DBDC4 Poster Session* (2019).
[5] Sugiyama, H.: Dialogue breakdown detection using BERT with traditional dialogue features, *DBDC4 WOCHAT workshop* (2019).
[6] Devlin, J., Chang, M.-W., Lee, K. and Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, Minnesota, Association for Computational Linguistics, pp. 4171–4186 (online), DOI: 10.18653/v1/N19-1423 (2019).
[7] 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 (2015). https://sites.google.com/site/dialoguebreakdowndetection/documents/.
[8] JongHo Shin, A. D. and Anshu, A.: Context-enriched Attentive Memory Network with Global and Local Encoding for Dialogue Breakdown Detection, *DBDC4 WOCHAT workshop* (2019).
[9] Mariya Hendriksen, A. L. and Moens, M.-F.: LSTM for Dialogue Breakdown Detection: Exploration of Different Model Types and Word Embeddings, *DBDC4 WOCHAT workshop* (2019).
[10] 桑原健太，大村英史，桂田浩一：Universal Transformer を使用した対話破綻検出，人工知能学会全国大会論文集，Vol. JSAI2019, pp. 4J3J1301–4J3J1301（オンライン），DOI: 10.11517/pjsai.JSAI2019.0_4J3J1301 (2019).
[11] Dehghani, M.，Gouws, S.，Vinyals, O.，Uszkoreit, J.，Łukasz Kaiser： Universal Transformers (2018).
[12] 稲葉通将，高橋健一：対話破綻検出の対話システムへの適用，人工知能学会論文誌，Vol. 34, No. 3, pp. B–I64 1–8（オンライン），DOI: 10.1527/tjsai.B-I64 (2019).
[13] 黒橋・河原研究室：BERT 日本語 Pretrained モデル - KUROHASHI-KAWAHARA LAB (2019). http://nlp.ist.i.kyoto-u.ac.jp/index.php?BERT%E6%97%A5%E6%9C%AC%E8%AA%9EPretrained%E3%83%A2%E3%83%87%E3%83%AB.
[14] hottolink: hottoSNS-w2v: 日本語大規模 SNS+Web コーパスによる単語分散表現モデル. https://github.com/hottolink/hottoSNS-w2v?_fsi=hzH8N6iT#%E5%88%A9%E7%94%A8%E8%A6%8F%E7%B4%84.
[15] 京都大学大学院情報学研究科黒橋・河原研究室：日本語形態素解析システム JUMAN++ version 1.01. http://nlp.ist.i.kyoto-u.ac.jp/index.php?JUMAN++.
[16] Loshchilov, I. and Hutter, F.: Decoupled Weight Decay Regularization (2017).