発話とレビューに対する解釈文生成とトピック分類

林部 祐太
株式会社リクルート Megagon Labs, Tokyo, Japan
hayashibe@megagon.ai

概要

我々は宿提案のための対話システムの構築に取り組んでいる．そこでは，カスタマーの発話意図をレビューなどから抽出した宿に関する情報とマッチングして，宿を提案する．それには，レビューやカスタマー発話内の文の意図を解釈する必要がある．そこで，文の意図を簡潔に示す「解釈文」を生成することで意図解釈を行う．そして，それぞれのトピックに基づきマッチングする．本論文では，解釈文生成とトピック分類の従来手法の改善に取り組んだ．

1 はじめに

我々は宿提案のための対話システムの構築に取り組んでいる．その対話システムは，単にカスタマーの入力した要望で検索するのではなく，宿レビューも参照しながら逆に質問して要望を詳細化していったり（詳細化），候補を絞るために異なる観点から要望を訪ねたり（要望追加），ときには代替案を提案したり（代替提案）と，インタラクティブに宿提案することを目的としている [1]．

そのためにはまず，レビューや発話内の文の意図を解釈する必要がある．例えば，システムが 2 択で質問した後，「後者がいい」とカスタマーが発話したとして，発話そのままでレビューなどを検索しても適切な結果は得られない．システムは「後者」が指す事柄を理解する必要がある．また，「すごく楽しめました。」というレビュー文では対象物が省略されているので，文そのままを検索対象にするのではなく，文の意図を適切に表した上で検索対象にする必要がある．そこで，文の意図を簡潔に表す「解釈文」にそれぞれ変換した上で，システム内で利用することにする．

そして，宿トピックに関する知識を用いてマッチングする．具体的には解釈文を事前に整備した宿トピックに関するツリー構造で整理した「トピックツリー」上に分類して，宿を提案する．例えば，『トピックの子トピックに「バイキング」や「定食」といった食事の形式に関するトピックがある』という知識を用いてカスタマー発話の解釈文「おいしい食事が食べたい」とレビューの解釈文「バイキングを楽しめた」に対して詳細化した宿を提案する．図 1 に解釈文生成とトピック分類に基づくマッチングのイメージを示す．

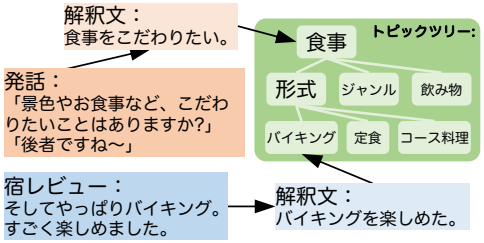

図 1 解釈文生成とトピック分類による発話とレビューのマッチング例（作例）

これまで，我々は発話に対する解釈文生成のためのコーパス構築とモデル学習 [2, 3] や，トピックツリーの整備と分類 [1] について取り組んだ．本論文では，

- 宿レビューへの解釈文アノテーション
- 解釈文の生成候補リランキングによる改善
- トピック分類のフィルタリングによる改善

についての取り組みについて報告する．

2 解釈文生成

2.1 解釈文コーパス

解釈文とは，発話理解の 1 つの形式で，対象の文の意味を文脈を読まなくても理解できるように表現した文 [3] である．[1)]

例を表 1 と表 3 に示す．対象の文の省略されている言葉を補ったり，照応詞を置き換えたりしている．また，表 1 の (2) や表 3 の (B) のように複文を単文に分けて表現することで，複雑な要望を理解・処

1) [2] では「要約」, [3] では“Self-Contained Utterance Description”とよんでるが，本論文では「解釈文」とよぶ

オペレータ	朝食のご希望はございますか？和食と洋食が選べます。またお部屋のご希望はございますか？
カスタマー	後者で。(1) 部屋は禁煙で海側だと嬉しいな。(2)
解釈文	朝食は洋食が良い。(1) 部屋が禁煙が良い。部屋は海側だと良い。(2)

表 1 発話と解釈文の例．解釈対象の文とその解釈文は，同一番号で下線を引いている．

	発話			レビュー	
	対話数	客側発話数	客側文数	文書数	文数
学習 (train)	32,769	40,685	47,275	380	1,721
開発 (dev)	5,431	7,175	8,450	48	224
評価 (test)	5,170	6,587	7,742	47	210

表 2 解釈文アノテーションの統計

理しやすくする．

本研究では，対話中の発話と，宿レビューの 2 種類を対象に解釈文コーパスを構築した．

2.1.1 発話の解釈文

架空の宿予約サービスにおけるカスタマーとオペレータの発話の作成を作業者に依頼し，対話を収集した．そして，各対話のカスタマーの発話の各文に対して，1 人のアノテータが解釈文を作成した．コーパスの統計を表 2 に示す[2)]．

2.1.2 宿レビューの解釈文

旅行情報サイト「じゃらん net」[3)] に投稿された宿レビュー 475 件[4)] にも解釈文を作成した．例を表 3 に示す．また統計を表 2 に示す．(A) の解釈文は，対象文では省略されている「横浜スタジアムに」や「ホテルから」を補い，文脈を見なくても対象の文の意図を理解できるようになっている．(B) では，複数の事柄が 1 文で書かれているレビュー文が複数の解釈文で表現され，システムの検索などで扱いやすくなっている．

2.2 解釈文生成モデルの構築

発話の解釈文を生成するモデルと，宿レビューの解釈文を生成するモデルの 2 つを構築する．モデルは [3] での実験と同様に，日本語プレトレーニング済 Text-to-Text Transfer Transformer (T5) [4] モデル[5)] を fine-tuning することで学習する．最適化アルゴリズムを Adafactor，学習率を 10^{-3}，エポック数を 20 とし，開発データでの Cross Entropy Loss が最も低いモデルを用いる．

2.3 解釈文候補のリランキング

構築した生成モデルは誤った解釈文を生成することがありうる．例えば，表 4 では第 1 解釈文候補には発話意図とは異なる数値が含まれており誤っている．このような正解との Cross Entropy Loss が小さくても致命的な誤りは，Cross Entropy Loss の最適化で学習したモデルでは除外が難しい．そこで，解釈文候補の正しさをスコア化するモデル（スコアラー）を別途用意し，複数の解釈文候補の中から尤もらしい候補を選ぶリランキングを行う．

スコアラーは日本語 RoBERTa モデル[6)] を fine-tuning することで構築し，正解らしさを 0 から 1 の確率値で予測する．fine-tuning には解釈文候補に対して正誤をアノテーションしたデータを用いる．

発話の解釈文候補のスコアラーは，学習に 7,769 件，開発に 3,821 件，評価に 775 件用いる．正例はそれぞれ 3,855 件，2,953 件，278 件含まれている．また，宿レビューの解釈文候補のスコアラーは，学習に 5,630 件，開発に 779 件，評価に 648 件用いる．正例はそれぞれ 1,998 件，283 件，213 件含まれている．

最適化アルゴリズムを AdamW，学習率を 10^{-3}，エポック数を 20 とし，開発データでの Cross Entropy Loss が最も低いモデルを用いる．

評価データでのスコアラーの性能は，スコアが 0.5 以上の事例を正解と予測したとすると，発話用が Precision=0.7231, Recall=0.9676, F1=0.8277 で，レビュー用が Precision=0.7034, Recall=0.7793, F1=0.7394 だった．

2.4 リランキングの効果検証

2 つのモデルはビーム幅 5 で探索して 5 つの解釈文候補を生成させる．そして，その候補の第 1 候補と，各候補にスコアラーが予測したスコアでリランキングした後の第 1 候補を比較して，リランキングの効果を検証する．具体的には，発話の解釈文生成

2) コーパスは [3] にて作成した 2 演者間の対話と “Additional Corpus” を含んでおり，一部を除き公開している: https://github.com/megagonlabs/asdc

3) https://www.jalan.net/

4) Evidence-based Explanation Dataset として公開しているレビューを用いた: https://github.com/megagonlabs/ebe-dataset

5) https://huggingface.co/megagonlabs/t5-base-japanese-web-8k

6) https://huggingface.co/rinna/japanese-roberta-base

(A)	レビュー	横浜スタジアムの野球観戦で利用しました。徒歩で行けたので良かったです。···
	解釈文	【customer】が横浜スタジアムにホテルから徒歩で行ける。
(B)	レビュー	··· 部屋からは海が見え、周辺は閑静で非日常的な空間を味わうことができました。···
	解釈文	【customer】が部屋から海が見える。周辺が閑静だ。【customer】が非日常的な空間を味わうことができる。

表 3 宿レビューと解釈文の例．例示対象の文に下線を引いている．

オペレータ	1 泊 1 室あたりの予算の上限はお決まりですか？
カスタマー	19999 円です
解釈文候補	(1) 1 泊 1 室あたりの予算の上限は 199 円だ。 (2) 1 泊 1 室あたりの予算の上限は 19999 円だ。 (3) 1 泊 1 室あたりの予算の上限は 20009 円だ。

表 4 第 1 候補が誤った解釈文となる例

評価データと，宿レビューの解釈文生成評価データの中から，リランキング前とのスコアの差が最も大きかった順にそれぞれ 35 事例を取り出し，改善の有無を人手で評価する．表 5 に結果の例を示す．

まず，発話の解釈文生成のリランキングでは，改善 11 件，悪化 15 件，変わらず 9 件だった．改善した事例では，欠落している情報が増えた事例 8 件，不要な情報が消えた事例 3 件だった．悪化した事例では，6 件がインフォーマルな表現になってしまった事例，必要な情報が消えた事例 5 件，不要な情報を付加してしまった事例 4 件だった．また，宿レビューの解釈文生成のリランキングでは，改善 16 件，悪化 4 件，変わらず 15 件だった．改善した事例の内訳は，欠落している情報が増えた事例 7 件，不要な情報が消えた事例 8 件，流暢性が増した事例 1 件だった．悪化した事例は，不要な情報が消えた事例 3 件，不要な情報を付加してしまった事例 1 件だった．

以上より，リランキングは生成の改善に一定の効果があることがわかった．しかし，十分な改善とはいえないため，さらなる改善を今後行う予定である．

なお，すべての出力に対してリランキングを行った場合の ROUGE[7] [5] スコアについて調べたところ，発話の解釈文生成では，リランキングで 0.867 から 0.790 に，宿レビューの解釈文生成では 0.775 から 0.749 に減少した．ROUGE では致命的な解釈の誤りを過小評価してしまうと考えられ，より適切なリランキング方法や評価指標の選定も今後の調査課題である．

	発話
解釈対象	両親ときょうだいと一緒に旅行したいと思っています。
正解	【customer】が両親ときょうだいと一緒に旅行したい。
前	【customer】が両親と一緒に旅行したい。
後	【customer】が両親ときょうだいと一緒に旅行する。（改善）
文脈	今度の旅行で泊る宿の部屋からの景気はこだわりますか？
解釈対象	ただ、そこまで綺麗な場所でなくても大丈夫です。
正解	部屋からの景色がそこまで綺麗な場所でなくても良い。
前	部屋からの景色が綺麗な場所でなくても良い。
後	そこまで綺麗な場所でなくても良い。（悪化）
	宿レビュー
解釈対象	とても清潔感がありキッズルーム、貸切風呂にはたくさんのおもちゃがありました。
正解	【宿】にとても清潔感がある。キッズルームにたくさんのおもちゃがある。貸切風呂にたくさんのおもちゃがある。
前	清潔感がキッズルームにある。貸切風呂にたくさんのおもちゃがある。
後	清潔感がとてもある。キッズルームにたくさんのおもちゃがある。貸切風呂にたくさんのおもちゃがある。（改善）
解釈対象	評価を参考にしこちらにお世話になりました。
正解	評価を参考にして【customer】が【宿】に世話になった。
前	【customer】が【宿】に世話になった。
後	(出力なし)（悪化）

表 5 解釈文生成結果のリランキングの前後の例

2.5 関連研究

解釈文生成のように後続の処理のため入力となる文を書き換える手法として，[6] が提案されている．リランキングに異なる分類器を用いた研究には，含意関係認識器を要約候補のリランキングに用いる手法 [7] がある．候補生成自体の改善には，学習する損失関数に負例も利用する手法[8] [8] や重複表現の生成を防ぐ手法[9] [9] などが提案されている．

7) SumEval の実装を用いた: https://github.com/chakki-works/sumeval

8) https://github.com/aistairc/contrastive_data2text

9) https://github.com/yxuansu/SimCTG

トピック	トピック例文
備品>アメニティ>アイマスク	アイマスクがある
立地>交通アクセス>鉄道	鉄道でのアクセスが良い
食事>アレルギー対応>そば	そばアレルギーに対応してくれる
風呂>基本>バス・トイレ付き>セパレート	部屋のお風呂とトイレが分かれている

表 6　宿トピックとトピック例文の例

3　解釈文のトピック分類

3.1　トピックツリーと分類モデル

[1] では，宿提案対話システムで用いる知識整備として，ツリー構造での宿に関するトピックを整理した．本論文ではそれを微修正したトピックツリーを実験に用いる．トピックツリーの例を表 6 に示す．

また，テキストのトピック分類には，同じく [1] で提案したトピック数が非常に多いが学習事例が非常に少ないという制約においても頑健に動く類似度に基づく手法を用いる．

3.2　既存モデルの問題点とフィルタリング

前述のトピック分類モデルは類似度に基づくため，類似はするが関係がないトピックを排除するのが難しい．例えば，「禁煙の部屋を希望する」という解釈文のトピックに「禁煙」のみならず「喫煙」も予測する，ということが起こりうる．また，「【customer】がパンを食べたい。」という解釈文のトピックに「パン」のみならず「クロワッサン」というトピックも予測しうる．

そこで，分類結果の正しさをスコア化するモデル（スコアラー）を別途用意し，閾値未満のスコアをもつトピックをフィルタリングすることで，関係のないトピックを排除する．

スコアラーはじゃらん net に掲載されている宿レビューを使って学習した BERT[10] を fine-tuning することで学習する．スコアラーはトピックを表す「トピック例文」と分類対象の解釈文のペアを入力として，正しさを 0 から 1 の確率値で予測する．なお，テキストのペアを入力とせずに，トピックごとに予測をする方法も考えられるが，トピック数が 1,051 と非常に多く，学習データを用意するのが困難であるため，その方法は用いない．分類問題に文の対を用いる手法としては，ゼロショットテキスト分類に含意関係認識を用いる手法 [10] が提案されている．

10）　分類モデルの学習で用いた BERT と同一

解釈文	前	後
夕食が和食のホテルが良い。	食事>ジャンル>和食>懐石料理	食事>ジャンル>和食（改善）
ペット用食事が提供されるホテルにする。	設備・サービス>ペットサービス>ペット宿泊可>ペットフード	設備・サービス>ペットサービス（改悪）
洗濯できるホテルを【customer】が希望する。	備品>耐久財>洗濯機	設備・サービス>フロント系サービス>クリーニング（どちらとも言えない）

表 7　トピック分類結果のフィルタリング前後の例

fine-tuning には発話の解釈文や宿レビューの解釈文のトピック分類結果に対して正誤をアノテーションしたデータを用いる．学習には，31,232 件，開発に 7,746 件，評価に 1,675 件用いる．このうち，正例はそれぞれ 20,722 件，5,859 件，241 件含まれている．

評価データでのスコアラーの性能は，スコアが 0.5 以上のトピックを正解と予測したとすると，Precision=0.6328, Recall=0.8797, F1=0.7394 だった．

3.3　フィルタリングの効果検証

既存モデルで評価用の解釈文にトピック分類モデルで 5 つの候補を類似度スコア付きで出力し，それぞれに対してスコアラーで「正しさスコア」を求めそれが 0.5 未満のトピックを候補から消去してフィルタリングする．残る候補の中で類似度スコアが最も高いトピックを新しい予測結果とする．

このフィルタリングによって，予測結果に変化があった 139 事例中 35 事例に対して評価した．内訳は，17 件が改善，7 件が悪化，11 件が改善とも悪化ともいえない，だった．例を表 7 に示す．以上より，多くの事例ではフィルタリングで改善できることが分かった．誤ったフィルタリングを減らすことや 1 つではなく複数のトピックに分類するよう拡張することなどが今後の課題である．

4　おわりに

本論文では解釈文生成とリランキングによる改善，およびトピック分類とフィルタリングによる改善について報告した．それぞれ改善する事例があるが，悪化する事例も少なくないため，さらなる手法の洗練が今後の課題である．

謝辞　アノテーションを行っていただき，多くの示

唆に富んだご意見をくださった山下華代氏に感謝します．また，有益な助言していただいた大阪大学の荒瀬由紀准教授に感謝します．

参考文献

[1] 林部祐太, Varga István. 宿トピックの整理と自動分類の試み. 言語処理学会年次大会, 2022.
[2] 林部祐太. 要約付き宿検索対話コーパス. 言語処理学会年次大会, 2021.
[3] Yuta Hayashibe. Self-contained utterance description corpus for Japanese dialog. In **LREC**, 2022.
[4] Colin Raffel, Noam Shazeer, et al. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. **Journal of Machine Learning Research**, Vol. 21, No. 140, 2020.
[5] Chin-Yew Lin and Eduard Hovy. Automatic evaluation of summaries using N-gram co-occurrence statistics. In **NAACL**, 2003.
[6] Zhiyu Chen, Jie Zhao, et al. Reinforced question rewriting for conversational question answering. In **EMNLP**, 2022.
[7] Tobias Falke, Leonardo F. R. Ribeiro, et al. Ranking generated summaries by correctness: An interesting but challenging application for natural language inference. In **ACL**, 2019.
[8] Yui Uehara, Tatsuya Ishigaki, et al. Learning with contrastive examples for data-to-text generation. In **COLING**, 2020.
[9] Yixuan Su, Tian Lan, et al. A contrastive framework for neural text generation. In **NeurIPS**, 2022.
[10] Wenpeng Yin, Jamaal Hay, et al. Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach. In **EMNLP**, 2019.