修士論文

複数の属性に対する評価を含む宿泊施設レビューに対する多様な返信の自動生成

村越 裕太

主指導教員 白井 清昭

北陸先端科学技術大学院大学
先端科学技術研究科
情報科学

令和 6 年 3 月

Abstract

With the wide spread of the Internet, nowadays, many people make reservation for accommodation at hotels via websites. Furthermore, on many online hotel reservation websites, users can post reviews about a hotel at which they stayed and hotels can reply to those user reviews. It is important for hotels to reply to user reviews in order to protect their brand image and enhance customer's satisfaction. However, replying to a large number of reviews requires vast time and resources, and replying to negative reviews can impose a substantial mental burden on a hotel staff. Therefore, it is required to develop a system that can automatically generate replies to hotel reviews.

Most previous studies on automatically generating replies to reviews focus on reviews about applications(software) and products, while generation of replies to hotel reviews has not been investigated much. In addition, previous studies do not pay attention to generation of replies that comprehensively address multiple aspects in reviews. Here "aspect" of a hotel refers to an element of a hotel such as "room", "bath", "staff" and other facilities, functions, and services, which is often evaluated in a review. If a user expresses dissatisfaction with multiple aspects but a hotel addresses only a few aspects or no aspect in a reply, the user may feel that the reply is insincere and get a negative impression on the hotel.

This study aims to generate appropriate replies to low-rated hotel reviews. Especially, the following two characteristics are heavily considered. First, when a user expresses dissatisfaction with multiple aspects of a hotel, the generated reply should address all these aspects. Second, generated replies should be diverse, not consist of only generic sentences in other words. Here, a generic sentence is defined as a sentence that only conveys general meaning and can be used in many reviews, such as "thank you." and "we are looking forward to your next visit." We aim to generate not generic sentences but specific sentences that are highly coherent with user's complaints.

The procedures of the proposed method are as follows. First, a hotel review is divided into individual sentences. Next, "complaint classification model" judges whether each sentence is a complaint, and sentences not identified as complaints are excluded. Then, the remaining complaint sentences are fed into "reply generation model" to generate a reply to each of the input sentences. Finally, the reply sentences generated for all complaint sentences are concatenated to produce the final reply to the review.

The complaint classification model is trained using Bidirectional Encoder Representations from Transformers(BERT). Reviews that consist of a single sentence and are labeled as a complaint are extracted from the Rakuten Travel dataset as positive samples, while the same number of single-sentence reviews without com-

plaint labels are randomly chosen as negative samples. These samples are used as the training data for fine-tuning of BERT.

The reply generation model is trained by fine-tuning the Bidirectional Auto-Regressive Transformer (BART), which is a sequence-to-sequence model. As the training data, pairs of user reviews and their corresponding replies written by hotels are obtained from the Rakuten Travel dataset. To improve the quality of the training data, two filtering methods are employed. The first filtering is to remove replies that do not address aspects in user reviews. Only pairs of a review and a reply where both contain the same aspect word are kept, while others are removed. This ensures that the reply generation model can produce replies that address the aspects appeared in the review. The aspect words of hotels are extracted from the user reviews in the training data in advance. TF-IDF scores of all words in all reviews are calculated, and the average of the top 1000 TF-IDF scores is used as the score of each word. The top 500 words ranked by this score are chosen as the aspect words.

The second filtering is to remove generic sentences from the training data. This process is expected to prevent the reply generation model from generating stereotyped replies. The detailed procedures of this filtering are as follows. Replies of hotels in the training data are divided into individual sentences, and a generic score is calculated for each sentence. This score is derived from the average of the frequency of all word tri-grams in the sentence, where the frequency is the number of occurrence of the word tri-gram in the training data. The top 30% sentences with the highest generic score are considered as generic and are removed from the training data.

In the last step of concatenation of replies to individual review sentences, duplicate replies are excluded. The similarity between the generated reply sentences is measured using a normalized edit distance. If this value is 0.1 or lower, the reply sentence that appears later in the original review is retained, and the other sentence is discarded. However, replies containing aspect words are always kept. After this procedure, the remaining reply sentences are concatenated in the same order as in the original review to form the final reply.

Several experiments were carried out to evaluate each of the complaint classification model and the reply generation model. To evaluate the complaint classification model, the model was applied for the classification of whether each review in the test data was a complaint, and the accuracy, precision, recall, and F1-score were measured. The best results were obtained when the number of training epochs was set to one, yielding 0.8877 accuracy, 0.8718 precision, 0.9091 recall, and 0.8901 F1-score. These results indicated that the performance of the complaint classification model was sufficiently high.

In the evaluation of the reply generation model, we compared several methods, such as methods with and without two filtering methods, and methods that generated replies to individual sentences and concatenated them or generated one reply to an entire review. A model fine-tuned from the training data without our filtering methods was used as a baseline. First, an automatic evaluation was conducted. The quality of the generated replies was assessed by BLEU-4 and DISTINCT-4. The BLEU-4 score of the baseline was higher than that of the proposed methods. The baseline tended to generate generic sentences, but the reference replies in the dataset also contained many generic sentences. Thus, many word n-grams were overlapped in the generated reply and reference, causing the BLUE-4 score to become high. Besides, the DISTINCT-4 score of the proposed method with filtering was higher than that of the baseline, indicating that our filtering methods were effective to suppress generation of generic sentences. Next, we conducted a human evaluation. Seven subjects rated the fluency, non-redundancy, overall quality, and aspect coverage rate (proportion of aspects appearing in reviews that are addressed in replies) of the generated replies. The results showed that the proposed method, which filtered out replies not addressing aspects, had a higher aspect coverage rate compared to the baseline, but its fluency and non-redundancy were worse. It indicates there is a trade-off between the aspect coverage rate and the fluency/non-redundancy. Filtering of generic sentences improved the non-redundancy, as well as the fluency and the overall quality. It means that the generation of stereotyped expressions is suppressed. In addition, for reviews containing multiple aspects, the technique to generate replies to not an entire review but individual sentences could improve both the aspect coverage rate and the overall quality, showing its effectiveness.

概要

インターネットの普及に伴い，宿泊施設の予約がウェブサイトを介して行われるようになった．また，多くのオンライン予約サイトで，ユーザによるレビューの投稿機能とそれに対する宿泊施設の返信機能が提供されている．ユーザのレビューに返信することは宿泊施設にとってブランドイメージを保護し，顧客満足度を高める効果がある．一方で，大量のレビューに対応することは多くの時間とリソースを要する作業であり，特に否定的なレビューへの返信はスタッフにかかる精神的な負担が大きい．このため，宿泊施設レビューへの返信を自動生成するシステムが求められている．

レビューに対し返信を自動生成する既存の研究の多くは，アプリケーションレビューや商品レビューを対象にしており，宿泊施設のレビューに対する返信に焦点を当てた研究はあまり多くない．加えて，これらの研究ではレビュー内の複数の評価対象の属性に対して網羅的に言及する返信の生成には留意していない．ここで宿泊施設の「属性」とは,「部屋」「風呂」「スタッフ」など，評価の対象となる宿泊施設の設備・機能・サービスなどを指す．ユーザが複数の属性に対して不満を表明しているにも関わらず，その一部についてだけ返信をすると，ユーザは不誠実な対応と感じ，宿泊施設に対して悪い印象を持つ可能性がある．

本研究は，宿泊施設に関する低評価レビューに対し，適切な返信を生成することを目的とする．この際，以下の 2 つの点に留意する．ひとつは，ユーザが宿泊施設に関する複数の属性に対して不満を表明しているとき，その全ての属性に言及することである．もうひとつは多様な返信の生成である．当たり障りのない一般的な表現を生成するだけでなく，ユーザの不満に応じた適切な表現を生成することを目指す．

提案手法の手順は次の通りである．初めに，レビューを個々の文に分割する．次に，苦情判定モデルにより，各レビュー文が苦情であるかを判定し，苦情でない文は排除する．続いて，苦情と判定されたレビュー文を返信生成モデルへ入力し，返信を生成する．最後に，各レビュー文に対して生成された返信を統合し，最終的な返信を出力する．

苦情判定モデルは，Bidirectional Encoder Representations from Transformers (BERT) により学習する．楽天トラベルのデータセットから，苦情ラベルが付与され，かつ単一の文から構成されるレビューを正例として抽出する．また，苦情ラベルが付与されていない単文のレビューを正例と同じ数だけランダムにサンプリングし，負例とする．これを訓練データとして BERT をファインチューニングする．

返信生成モデルは，系列変換モデルである Bidirectional Auto-Regressive Transformer(BART) をファインチューニングすることで学習する．訓練データとして，楽天トラベルのデータセットから取得したレビューとそれに対する宿泊施設の返信の組を用いる．また，訓練データの品質を高めるために，2 つの訓練データのフィルタリング手法を採用する．1 つ目は，属性に言及しない返信のフィルタリングである．レビューと返信の両方に属性語が存在する組のみを残し，それ以外は

削除する．これにより，返信生成モデルがレビュー内の属性に言及した返信を生成できるようにする．属性語は訓練データから事前に抽出する．全ての単語の全てのレビューにおける TF-IDF を算出し，その TF-IDF 値の上位 1000 件の平均を単語のスコアとし，そのスコアの上位 500 件の単語を属性語とする．2 つ目は，定型文のフィルタリングである．定型的な表現で書かれた返信文を訓練データから削除する．これにより，モデルが紋切り型の返信を生成することを抑制することが期待できる．処理の手順は次の通りである．訓練データの返信を文に分割し，それぞれの返信文に対して定型度スコアを算出する．定型度スコアは，返信文における単語の tri-gram の訓練データ全体における出現頻度の平均値として算出する．スコアが最も高い上位 30%の文を定型文とみなし，これらを訓練データから削除する．

返信の統合では，文毎に生成された返信をマージする．この際，重複する返信文を除外する．生成された返信文間の類似度を正規化された編集距離で測定し，この値が 0.1 以下の場合，元のレビューにおける出現順序が後の返信文を保持し，他方を削除する．ただし，属性語を含む返信については除外しない．この手続きの後に残された返信文を，元のレビューと同じ順序で連結し，最終的な返信とする．

提案手法の評価実験では，苦情判定モデルと返信生成モデルをそれぞれ評価した．苦情判定モデルの評価では，テストデータにおけるレビュー文が苦情か否かを分類し，その正解率，精度，再現率，F 値を測った．その結果，訓練時の epoch 数を 1 に設定したときの結果が一番良く，そのときの正解率は 0.8877，精度は 0.8718，再現率は 0.9091，F 値は 0.8901 であった．これにより苦情判定モデルの性能が十分に高いことを確認した．

返信生成モデルの評価では，訓練データに対するフィルタリングの有無や，文毎に返信を生成するかどうかなどの条件を変えた手法を比較した．また，訓練データに対するフィルタリングを行わずに返信生成モデルを学習する手法をベースラインとした．まず，自動評価を行った．生成された返信の品質を BLEU-4 と DISTINCT-4 で評価した．BLEU-4 は提案手法よりもベースラインの方が高かった．これは，ベースラインは定型文をよく生成するが，正解の返信にも定型文が多く，単語 n-gram の一致率が高くなったためと推測される．一方，DISTINCT-4 は訓練データのフィルタリングを行った提案手法で高くなり，定型的な表現の生成の抑制にフィルタリングが有効であることを示した．次に，人手による評価を行った．返信の流暢性，非冗長性，総合評価，属性言及率 (レビューに出現する属性のうち返信で言及されているものの割合) を 7 名の被験者が評価した．その結果，属性に言及しない返信のフィルタリングを行う提案手法は，ベースラインに比べて属性言及率が高くなる一方で，流暢性と非冗長性は低くなった．これは，属性言及率と流暢性・非冗長性がトレードオフの関係にあることを示唆する．また，定型文のフィルタリングに焦点を当てると，非冗長性が改善されるとともに，流暢性や総合評価も高くなり，紋切り型の表現の生成が抑制されていることが確認された．さらに，複数の属性を持つレビューに対しては，文毎に返信を生成する手法が，そうでな

い手法に比べて属性言及率と総合評価で優れており，その有効性が確認された．

目 次

図 目 次

表 目 次

第1章　はじめに

1.1　背景

近年，インターネットの普及に伴って宿泊施設の予約もウェブサイトを通じて行うことが増えてきている [5]．そのような宿泊施設のオンライン予約サイトの中にはユーザがレビューを書く機能がある．さらに，宿泊施設がレビューに対して返信できる機能も備わっている場合が多い．ユーザのレビューに返信することで，宿泊施設がユーザの声に耳を傾けていることをアピールできると同時に，否定的なレビューに対して迅速な対応を行うことで悪い評判が広まることを最小限に抑え，潜在的な顧客への悪影響を軽減し，ブランドイメージを守ることができる．また，個々のレビューに返信することで，顧客が個別に認識され，宿泊施設によって大事に思われていると感じさせることができる．これにより顧客の満足度は高まり，リピーターを増やすことにもつながる．一方で，宿泊施設が継続的にレビューに返信することは，大きな負担であることも事実である．レビューへの返信は一度きりの対応ではなく，特に人気のある施設や大規模なホテルチェーンでは毎日大量のレビューが投稿されるため，それらすべてのレビューに目を通し，適切な返信をすることは，非常に時間のかかる作業である．また，ユーザレビューに対して適切な返信を行うには，顧客サービスに精通し，宿泊施設の声を体現できる訓練されたスタッフが必要であり，このために大きなリソースを必要とする．また，否定的なレビューへの対応はスタッフにかかる精神的な負担が大きい．これらの課題を解決するために，レビューへの返信を自動生成することで，宿泊施設の負担を軽減することが求められている．

レビューに対して返信を自動生成する試みは，大規模言語モデルを利用する方法が主流となっている．しかし，それらの多くは製品やアプリケーションのレビューに対して返信を生成することを目的としており，宿泊施設のレビューに対する返信に焦点を当てた研究はあまり多くない．また，それらの研究の中でも，定型的な表現で単に謝罪をしているだけでないか，宿泊者が表明した複数の不満の全てに言及しているかといった観点から，返信の質を十分に検討した研究は存在しない．そのため，上記のような返信の質も考慮した上で自動返信システムを構築することが求められる．

1.2 目的

本研究は，宿泊施設に関する低評価レビューに対し，それに対する適切な返信を生成することを目的とする．この際，以下の 2 つの点に留意する．ひとつは多様な返信の生成である．生成モデルは当たり障りのない一般的な表現を生成する傾向がある [10] が,「申し訳ありません」「ご迷惑をおかけしました」など謝罪を表す定型的な表現を生成するだけでは，ユーザは宿泊施設が事務的な対応をしているという印象を持ち，ユーザの不満が解消されない可能性がある．もう一つは，ユーザが宿泊施設に関する複数の属性に対して不満を表明しているとき，その全ての属性に言及することである．ここで宿泊施設の「属性」とは,「部屋」「風呂」「食事」「スタッフ」など，評価の対象となる宿泊施設の設備・機能・サービスなどを指す．これら 2 つの目的を達成するため，ユーザレビューを入力，返信を出力とする系列変換モデルを学習するが，定型的な表現の抑制やレビュー中の属性に対する網羅的な言及を実現するための手法を提案する．

定型的な表現を抑制したりレビュー中の属性に言及した返信を生成したりすることの重要性について，例を挙げて説明する．表 1.1 と表 1.2 は実際のレビューとそれに対する宿泊施設の返信である．表 1.1 は本研究が目指すべき返信例であり，表 1.2 は本研究で適切ではないと考える返信例である．表 1.1 において，ユーザは「ベッドサイドのテーブルの埃」「メモ帳」「電気の明るさ」の 3 つの不満に言及し，宿泊施設もそれぞれの不満に対して適切な謝罪と解決策を提示している．一方，表 1.2 では，ユーザは「浴場の寒さ」について言及しているにも関わらず，宿泊施設の返信はそのことには触れていない．また，返信は丁寧で長いものの，どのようなレビューの返信にも当てはまるような定型的な表現だけから構成されている．このように，当たり障りのない表現でユーザに対する不満をあたかも無視しているような返信は，ユーザにポジティブな印象を与えるどころか，かえって不誠実さを感じさせる可能性がある．したがって，定型的ではなく多様な表現によって，レビュー中に表明されたユーザの不満に適切に対応した返信を生成することは重要な課題である．

1.3 本論文の構成

本論文の構成は以下の通りである．まず，2 章では，本研究に関連する研究を紹介し，これらに対する本研究の特徴を述べる．3 章では，本研究で提案する宿泊施設レビューに対する返信の生成方法を説明する．4 章では，提案手法の評価実験の手順，結果及び考察について述べる．最後に，5 章で本研究のまとめと今後の課題について述べる．

表 1.1: 適切な返信の例

レビュー	返信
駅からも徒歩圏内ですし、朝食つきですのでよかったと思います。トリプルの部屋に泊まりましたが、(しばらく使用してなかったのかと思うくらい）ベッドサイドのテーブルに埃が目立っていました。メモ帳も最初から設置していないのでしょうか。電気も明るさが調節できればよかったと思います。全体的に薄暗い印象を覚えました。	この度はご宿泊いただき誠にありがとうございます。ご指摘いただいた清掃不備につきまして、不愉快な思いをさせてしまい誠に申し訳ございませんでした。今後清掃担当者への指導を実施し、清掃強化を図って参ります。またメモ帳につきましては、客室内にご用意させていただいておりますが、今後お客様が使用しやすい場所への設置変更を検討して参ります。客室電気につきましても新たな照明機器の導入を検討して参ろうと考えます。貴重なご意見を頂戴致しまして誠にありがとうございます。今後もお客様にご満足いただける商品とサービスが提供できるよう、従業員一同努力をして参ります。次回名古屋へお越しの際にも当ホテルをご利用いただければ幸いでございます。従業員一同、次回のご宿泊を心よりお待ち申し上げます。

表 1.2: 不適切な返信の例

レビュー	返信
浴場が寒い。	この度はご利用いただきまして誠にありがとうございました。また、ご指摘に件につきましては大変申し訳ございません。貴重な御意見としてお受け致します。また宜しければご利用を心よりお待ち致しております。

第2章　関連研究

本章では，本論文の関連研究について述べる．2.1 節では，提案手法で利用する言語モデルを紹介する．2.2 節では，宿泊施設レビューに対する返信を生成する研究について述べる．2.3 節では，アプリケーションや商品に関するレビューに対する返信を生成する研究について紹介する．2.4 節では，接客業レビューとアプリケーションレビューの違いを論じた研究を紹介する．2.5 節では，先行研究と本研究の違いについて論じる．

2.1　言語モデル

本節では，提案手法で利用する既存の言語モデルの概要を紹介する．

2.1.1　BERT

Bidirectional Encoder Representations from Transformers(BERT)[6] は，双方向 Transformer[18] を用いた大規模な言語表現モデルであり，文脈に依存した単語の埋め込みの生成，文の分類問題，文間関係の分類問題などに応用できる．BERT の学習は，事前学習とファインチューニングの 2 段階の手続きで構成される．

BERT の事前学習では，タスク固有のデータを用いてモデルを学習する前に，大規模なテキストコーパスを用いて言語の一般的な抽象表現を学習する．この段階では，モデルは「Masked Language Model」(MLM) と「Next Sentence Prediction」(NSP) を用いて訓練される．MLM タスクでは，入力テキストの一部の単語がランダムに [MASK] という特殊なトークンにより置換され，そのマスクされた単語を文脈から予測するようにモデルが訓練される．これにより，BERT は左右の文脈を考慮して単語の意味を理解し，双方向の深い文脈表現を学習する．NSP タスクは，与えられた二つの文が連続して出現するかどうかを予測する．このタスクにより，モデルは文間の関係を理解し，文よりも広い文脈を考慮した抽象表現を学習する．

BERT のファインチューニングは，事前学習されたモデルを特定のタスクに適応させるプロセスである．この段階では，特定のタスクのデータセットを用いて，モデルの全パラメータを調整する．ファインチューニングは，通常以下の手順で行われる．まず，大規模なテキストコーパスで事前学習された BERT モデルをロー

ドする．次に，特定のタスクに対応するため，入力データのフォーマットを BERT の入力に合わせて変換し，タスクに応じた出力層を追加する．たとえば，感情分析タスクでは，ポジティブまたはネガティブの 2 つのクラスを出力とする全結合層を出力層として追加する．その後，タスク固有のデータセットを用いて，事前学習されたモデルのすべてのパラメータを調整する．

2.1.2 BART

Bidirectional Auto-Regressive Transformer(BART)[12] は，様々な自然言語処理タスクに応用できる sequence-to-sequence の大規模自然言語モデルである．BART のアーキテクチャは，BERT[6] のような双方向のエンコーダモデルと，GPT (Generative Pre-trained Transformers)[16] のような自己回帰のデコーダモデルを組み合わせた Encoder-Decoder モデルであり，大まかな構造は Transformer[18] と同じである．BART の学習は，BERT と同様に事前学習とファインチューニングという 2 段階で構成される．

事前学習では，大量のラベルなしテキストから言語の抽象的な意味表現を学習する．事前学習は，図 2.1 のように，何らかの方法で破損させたテキストを双方向のエンコーダの入力として，自己回帰型デコーダのモデルにより文を出力し，その出力と元の文とのクロスエントロピーを最適化することで実現する．すなわち，破損した文を再構築することで，汎用言語モデルを学習する．文献 [12] では文の破損方

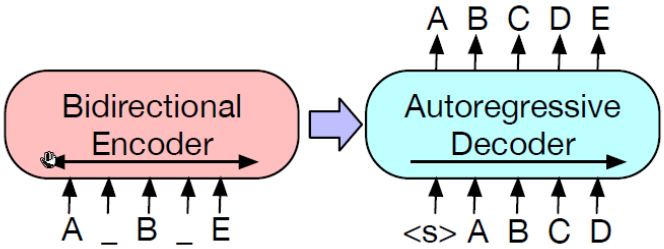

図 2.1: BART の事前学習 (文献 [12] より引用)

法として，図 2.2 のように，単一の単語をマスクする Token Masking，トークンをいくつか削除する Token Deletion，文の順番を入れ替える Sentence Permutation，複数単語列を単一トークンでマスクする Text Infilling，ランダムに選んだ単語が一番初めになるように文章を回転させる Document Rotation の 5 つの手法が提案されている．

BART のファインチューニングは，BERT と同様に，特定のタスクに対するモデルの性能を向上させるために，ラベル付きデータや生成元と生成後の文の組からなるデータを用いて，事前学習済みのモデルのパラメータを調整する処理を指す．例えば，文書分類タスクを BART で解く場合，分類したい文章をエンコーダ

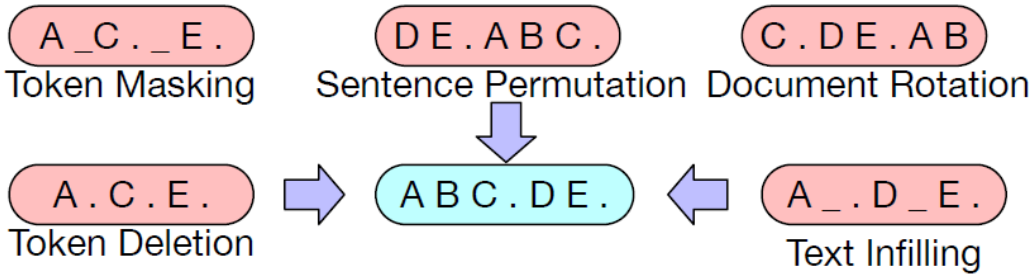

図 2.2: BART の事前学習における文の破損方法 (文献 [12] より引用)

の入力とデコーダの出力として与え，最後のトークンのデコーダにおける埋め込みを入力，分類クラスを出力とする線形分類器を BART につなげ，BART ならびに線形分類器のパラメタを更新する．文章要約タスクを解く場合は，要約対象の文書をエンコーダの入力とし，それに対する要約をデコーダが出力するようにモデルのパラメタを更新することで，要約に特化したモデルを学習する．

2.2 宿泊施設レビューに対する返信生成に関する研究

Kew と Volk は，オンライン上の宿泊施設レビューへの言語モデルによる返信生成能力を向上させることを目的として，言語モデルが利用する訓練データをフィルタリングする手法を提案した [10]．レビューと返信の組からなるデータセットから返信生成モデルを学習すると，多くのレビューに対して用いられる汎用的な表現を含む返信が生成されやすいという問題に対処するため，訓練データにおける返信の汎用性のスコアを算出し，そのスコアが閾値以上の返信を訓練データから除外した．この手法は,「訓練データで頻繁に観察される一般的な返信は，モデルに有益でない返信を学習させ，より具体的な返信を生成する能力を低下させる」という仮説に基づいている．

Kew らは，3 つのフィルタリング手法を提案している．1 つ目は語彙頻度を利用する方法 (Lex. freq.) で，返信テキストを Bag of Words として $\{w_1, w_2, ..., w_m\}$ と定義し，汎用的で出現頻度が高く，具体的な情報を伝えていないと考えられる単語を識別する．具体的な計算方法は式 (2.1)，(2.2) の通りである．

$$S_{lex\_freq} = \frac{\sum_{i=1}^{m} L(w_i)}{m} \tag{2.1}$$

$$L(w_i) = \begin{cases} 1 & if\ count(w_i, T) \geq t \\ 0 & otherwise \end{cases} \tag{2.2}$$

ここで，T は訓練データ内の返信の集合，$count(w_i, T)$ は訓練データの全ての返信における単語 w_i の出現頻度である．データセット全体における出現頻度が大きい

(閾値 t 以上の) 単語が多く出現するような返信は，スコア S_{lex_freq} が高くなる．2 つ目の手法は文レベルで返信の一般性を測定する方法 (Sent. avg.) である．性能の低い生成モデルから生成される返信のすべてを一般的な文例のプール $\mathcal{G}$ とし，各返信がそれらの文例のプールとどの程度意味的に類似しているかを計算する．計算式を式 (2.3)，(2.4) に示す．

$$\xi(s) = \max_{g \in \mathcal{G}}(\cos(s, g)) \tag{2.3}$$

$$\mathcal{S}_{sent\_avg} = \frac{1}{n}\sum_{i=1}^{n}\xi(s_i) \tag{2.4}$$

3 つ目は文の perplexity を利用する方法 (LM PPL) である．文書レベルで返信の一般性をスコアリングするために，Causal Language Model(CLM) を使用し，各返信の perplexity を計算する．頻出する一般的な返信は意外性が低いため，特異性の高い返信とは対照的に，CLM の perplexity は低くなる．実験では，ドメインに合わせてファインチューニングされた GPT-2[17] を用いて perplexity を算出している．また，フィルタリングしたデータから返信生成モデルを学習する系列変換モデルとして，BART[12] を採用している．

評価実験の結果，3 つの手法の全てについて，3 フィルタリングされたデータセットで学習されたモデルは，フィルタリングをせずにデータセット全体で学習されたベースラインモデルに比べて性能が向上した．実際の返信と生成された返信を比較した ChrF メトリクスは，ベースラインが 30.47 であるのに対し，Lex.freq. は 33.6，Set.avg. は 32.53，LM PPL は 32.63 となり，いずれもベースラインを上回った．元のレビューと返信を比較した ChrF においても，ベースラインが 15.87 であるのに対し，Lex.freq.，Set.avg.，LM PPL はそれぞれ 20.63，20.2，21.0 となった．Self-BLEU を指標とした返信の多様性の評価では，LM PPL が最も高く，Self-BLEU は 4.24 となったが，それでも GOLD(実際の返信) の 1.18 には及ばなかった．返信の長さはベースラインが 35.91 と最も短く，汎用的な文が多いことがわかった．Lex.freq. の手法で学習されたモデルが生成する返信の平均文長はもっとも長く，82.37 という結果になった．

鳥海と伊草は，RNN(Recurrent Neural Network) に基づく Encoder-Decoder モデルを用いて，ユーザのレビューから適切な返信を自動生成するシステムを提案した [20]．また，ユーザ評価や返信の長さといったレビュー特性を追加することで，返信の品質を向上させる方法についても検討した．346 万件以上のレビューと返信のペアからなる楽天トラベルのデータを用いた評価実験を行った．レビュー特性を加えたモデルは標準のモデルに比べて，単語の正解率と BLEU スコアが共に向上することを確認した．これにより，ユーザ評価と返信の長さが返信生成モデルの精度向上に寄与することを示した．しかし，実装したモデルはいずれも実用的なレベルには達しておらず，さらなる改良が必要であると結論付けている．

2.3 アプリ・商品レビューに対する返信生成に関する研究

Gao らは，アプリストアのユーザレビューに対する返信を自動生成するシステムである RRGen を提案した [8]．RRGen は，レビューと返信のペアでファインチューニングした基本的な RNN の Encoder-Decoder モデルに，アプリのカテゴリ，レビューの長さ，ユーザ評価，センチメントの 4 つのレビュー特有の特徴を attention 機構により組み込んだものである．これらのレビューの特徴を組み込むことで，重要なユーザの感情やレビュー内の主なトピックを捉えられるようになり，適切な返信を生成することができるようになる．例えば，1 つ星評価や長文のレビューでは，5 つ星評価や簡潔なコメントよりもモデルが謝罪を表す返信を生成しやすくなる．RRGen モデルは，279,792 のレビューと返信のペアのデータセットで学習された．評価実験では BLUE-4 スコアに基づく自動評価と，返信の流暢さ，レビューとの関連性，正確さの人手評価の両方が行われた．その結果，RRGen はベースラインモデルを大幅に上回ることが確認された．特に BLUE-4 スコアでは，ベースラインモデルが 21.61 であるのに対し，RRGen は 36.17 であり，その差は 14.56 ポイントであった．提案手法でモデルに組み込んだ 4 つのレビューの特徴について，その効果を Ablation Test により検証し，その全てがモデルの性能改善に貢献すること，全ての特徴を組み込んだモデルが最高の成績を上げたことを示した．人手評価でも，GOLD である公式開発者の返信が一般的に好まれたが，RRGen はベースラインモデルよりも全ての評価指標で高いスコアを出し，特に返信の流暢さの点で良い結果を残した．

Zhao らは，電子商取引のプラットフォームにおけるユーザレビューへの対応を自動化するアプローチを提示した [19]．ベースは Sequence-to-Sequence（Seq2Seq）の深層学習モデルであり，そこに gated multi-attention メカニズムと copy メカニズムを通じて商品情報を取り込む仕組みを提案した．gated multi-attention メカニズムは，モデルが出力の各単語を生成する際に入力テキストの異なる部分に注意 (attention) を当てることを可能にする．この場合，モデルはレビューの内容と特定の製品の詳細の両方に attention を当て，生成される返信が文脈を適切に反映できるようになる．copy メカニズムは，モデルが入力テキストから出力に直接単語をコピーすることを可能にし，特定の製品名や専門用語，ユーザのレビュー固有の表現を返信に含める場合に効果を発揮する．

評価実験の結果，提案モデルは全ての自動評価の評価基準においてベースラインを上回り，多様で有益な返信を生成することを確認した．特に，ROUGE-L，BLEU，及び Distinct-2 のスコアは，製品情報を利用した Copynet の拡張モデルと比較して，大幅に向上した．また，人間による評価では，5 人の被験者が，無作為に選ばれた 100 のレビューに対し，異なるモデルによって生成された返信を評価した．被験者は返信を 0 から 2 の尺度で採点し，0 はレビューに無関係または流暢でない，1 は関連はあるが十分な情報がない，2 は関連がありかつ十分な情報を持つ，とし

た．提案モデルは，2 と評価された返信の割合が最も高く，返信の大半がレビューと関連性があり，かつ十分な情報を含むという意味で有益であることを示した．これは，製品情報を利用した Copynet モデルを含む他のモデルを大きく上回っており，人間の視点から見た実用的な有効性を示している．また，人間が生成した返信は依然としてモデルを上回っているが，僅差であり，モデルの性能が高いことを確認している．

2.4 アプリケーションレビューと接客業レビューの差異に関する研究

Kew らは，飲食店と宿泊施設に対するレビュー (接客業レビュー) への返信を自動生成するシステムの性能と課題を調査した [9]．Gao らによって提案されたアプリケーションレビューへの返信を生成する seq2seq モデル [8] を，接客業レビューに対する返信の自動生成に適用した．このモデルはアプリケーションレビューの返信生成において一定の成功を収めていたが，接客業の分野に適用するとパフォーマンスが大幅に低下した．このようなアプリケーションレビューと接客業レビューの違いについて，前者はテキストが短く特定の機能や問題に焦点を当てているのに対し，後者はテキストが長く詳細で，サービス，立地，清潔さなど滞在中に直面する様々な側面について論じる傾向が見られると分析した．そして，接客業の領域におけるレビューの返信の自動生成はより難しいタスクであると述べ，それに関する独自の課題を指摘している．

2.5 本研究の特色

レビューに対する返信生成の先行研究は，アプリケーションレビューや商品レビューを対象とした研究が多く，宿泊施設レビューの領域での返信生成に関するものは少ない．また，2.4 節でも述べたように，アプリケーションレビューで成功した手法を用いても，それを宿泊施設レビューに適用すると大幅にパフォーマンスが低下することも指摘されている．

それに加え，先行研究ではユーザレビューにおける複数の評価対象の属性に対して，その全てを網羅的に言及する返信を生成することは留意されていない．本研究では，宿泊施設の低評価レビューについてはユーザの不満の全てに言及することが重要と考え，それを実現する方法を探究する．また，Kew と Volk の手法 [10] を参考に，当たり障りのない表現の生成を抑制することにも取り組む．

また，本研究は先行研究とは異なり，星の数によるユーザの宿泊施設に対する評価など，レビュー以外の情報を利用しない．一般に，利用可能な情報はレビューサイトによって異なるが，提案手法はレビューのみを入力とするため，どのようなサイトのレビューに対しても返信を自動生成できるという利点がある．

第3章　提案手法

3.1　概要

提案手法の概要を図 3.1 に示す．まず，句点を文境界として，レビューを文に分割する．次に，苦情判定モデルを用いて，個々の文が苦情か否かを判定し，苦情と判定されなかった文を削除する．苦情を含む分割されたレビュー文に対し，返信生成モデルを用いて宿泊施設の返信を生成する．最後に，生成された返信文を統合して，最終的な返信を生成する．

以下，文分割に関しては 3.2 節，苦情判定モデルの学習方法に関しては 3.3 節，返信生成モデルの学習方法に関しては 3.4 節，生成文の統合処理に関しては 3.5 節で，それぞれ詳細を述べる．

レビューに対して返信を生成する先行研究の多くは，レビューを入力，返信を出力とする End-to-End の系列変換モデルを学習するのに対し，本研究では，まずレビューを文に分割し，それぞれの文に対し返信を生成する．この狙いはレビューで言及されている複数の属性に対して漏れなく返信することにある．詳しくは 3.2 節で述べる．

返信生成モデルならびに苦情判定モデルの学習には楽天データセット [14] における楽天トラベルのデータ (以下,「楽天トラベルデータセット」と呼ぶ) を用いる．同データセットは宿泊施設に対するユーザレビューとそれに対する宿泊施設の返信を含む．また，ユーザレビューには「感想・情報」「苦情」などのラベルが付与されている．データセットの詳細については 4.1 節で述べる．

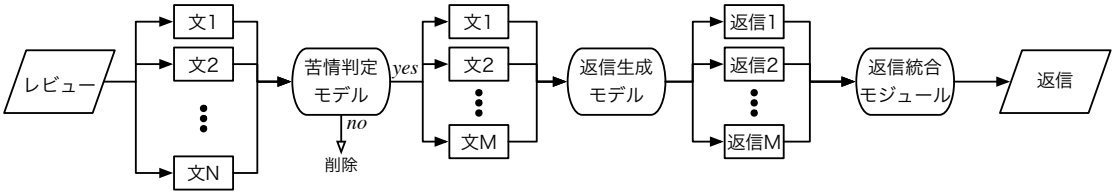

図 3.1: 提案手法の概要

3.2　文分割

レビュー全体を入力として返信を生成すると，複数の苦情を含むレビューに対しては，その全てが言及されない可能性が高いと考えられる．そのため，複数の

属性は異なる文で言及されていると仮定し，まずレビューを文に分割した後，それぞれの文から属性に対する言及を含む返信を生成する．これにより，レビュー内の複数の属性 (苦情) に対して網羅的に返信することを狙う．

文毎に返信を生成することの利点を具体例を挙げて説明する．図 3.1 は複数の観点から不満を述べているレビューと，そのレビュー全体を入力として，3.4 節で後述する方法で学習した系列変換モデルで返信を出力した例である．レビューでは，「風呂の水はけ」「風呂の換気扇」の 2 つの属性について不満が述べられているが，レビュー全体を入力とするモデルによる返信では，2 つ目の部屋の風呂の換気扇についてしか言及されていない．

表 3.1: 複数の属性に言及するレビューと自動生成した返信の例

レビュー	返信
駅から近いが、お風呂の水はけが悪かった。またお風呂の換気扇の音が電気を消してから相当たってから大きな音で「ビューン」といって切れるので、睡眠の妨げになった。	お風呂の換気扇の音につきましては、大変ご迷惑をお掛け致しました。早急に原因を究明致します。またのご利用を心よりお待ち申し上げております。

同じレビューを文に分割し，それぞれの文から返信を生成した例を表 3.2 に示す.「風呂の水はけ」と「風呂の換気扇」はそれぞれ 1 番目と 2 番目の文で言及されており，個々の文に対する返信ではそれぞれの属性の不備について謝罪している．このように，レビューを文に分割し，それぞれの文から個別に返信を生成することで，全ての属性が言及される可能性を高めることができる．

表 3.2: 文分割と返信の自動生成の例

文に分割されたレビュー	返信
駅から近いが、お風呂の水はけが悪かった。	お風呂の水はけが悪かったとのことで、大変申し訳ございませんでした。今後このようなことがないよう、清掃スタッフに指導して参ります。またのご利用をお待ち申し上げております。
またお風呂の換気扇の音が電気を消してから相当たってから大きな音で「ビューン」といって切れるので、睡眠の妨げになった。	お風呂の換気扇の音につきましては、大変申し訳ございませんでした。今後このようなことのないよう、点検を徹底して参ります。またのご利用を心よりお待ち申し上げております。

文分割は，句点を「。」「.」「.」「！」「？」「!」「？」のいずれかとして，句点で文を区切ることにより処理する．ただし，句点が鉤括弧 (「」『』) の中に存在する

場合は文を区切る記号として利用しない．表 3.3 はレビューの例であり，表 3.4 は表 3.3 のレビューを文に分割したものである．鉤括弧の中にある句点として，文 s_2 には「？」が，文 s_4 には「。」が，それぞれ存在する．しかし，これらは鉤括弧の中に存在しているため，文を区切る記号としては利用していない．

表 3.3: レビュー例

12 時 OUT にしてるのに 10 時にチェックアウトの電話がきた。しかも楽天経由で予約しているにもかかわらず、「電話でご予約ですよね？」とフロントは強気。プランを見てくださいと言って初めてやっと謝り始めた。「すみませんでした。失礼します。」とすぐに電話を切ろうとする。12 時までゆっくりしようと思ったが気分がわるくなった。

表 3.4: 表 3.3 のレビューの分割

ID	レビュー文
s_1	12 時 OUT にしてるのに 10 時にチェックアウトの電話がきた。
s_2	しかも楽天経由で予約しているにもかかわらず、「電話でご予約ですよね？」とフロントは強気。
s_3	プランを見てくださいと言って初めてやっと謝り始めた。
s_4	「すみませんでした。失礼します。」とすぐに電話を切ろうとする。
s_5	12 時までゆっくりしようと思ったが気分がわるくなった。

3.3 苦情判定

本研究では苦情に対する返信を生成することを主たる目的としているため，苦情を含まない文をあらかじめ削除する．これは，全体としては苦情を述べているレビューの中にも，文単位で見ると，苦情ではない文が存在するためである．以下の図 3.5 はそのような苦情を述べていない文の例である．全体としては宿泊施設の部屋の壁が薄く，隣の話し声が聞こえてしまうという苦情ではあるが，1 文目「駅からも近く、... よかった。」は苦情ではなく宿泊施設の良かったところについて言及している．この場合，苦情に対する返信を生成するときには，1 文目は答える必要のない文であるため，返信生成モデルの入力から除外するのが望ましい．

このため，文がユーザの苦情を含むか否かを判定する二値分類器を学習する．分類モデルとして BERT[6] を用いる．具体的には東北大学が公開している BERT base Japanese[2] をファインチューニングする．訓練データとして，楽天トラベルデータセットにおける「苦情」ラベルが付与されたレビューを正例，それ以外のレビューを負例として用いる．楽天トラベルデータセットでは,「苦情」ラベルが付与されたレビューの数はそうでないレビューの数よりもはるかに少ない．同デー

表 3.5: 苦情を表す文と表さない文の両方を含むレビュー

駅からも近く、長期の連泊で素泊まりでしたが周辺に食事するところも多くよかった。部屋もきれくスタッフの対応もよかったのですが、部屋の壁が薄いようです。隣の話し声が聞こえやすく、酔っ払って大きな声で話す客がいたときは迷惑でした。

タセットをそのまま訓練データとして用いると，判定が負例に偏るモデルが学習される可能性が高い．そのため，負例をランダムにサンプリングし，同数の正例と負例からなる訓練データを構築する．

また，楽天トラベルデータセットでラベル付けされているのは複数の文から構成されるレビューであるのに対し，提案手法における苦情判定の対象は文である．このため，楽天トラベルデータセットにおけるレビューのうち 1 つの文から構成されるレビュー (句点が 1 つしかないレビュー) のみを訓練データとして用いる．

3.4 返信生成モデル

苦情を含むと判定されたレビュー文に対し，それに対する宿泊施設の返信を生成する．このため，レビュー文を入力，宿泊施設の返信を出力とする系列変換モデルを学習する．系列変換モデルとして BART[12] を利用し，日本語事前学習済み BART モデル [1] をファインチューニングする．ファインチューニングのための訓練データとして，楽天トラベルデータにおける「苦情」のラベルが付与されたレビューとそれに対する返信の組を用いる．ただし，1.2 節で述べた我々が望ましいと考える返信を生成するため，訓練データに対して 2 種類のフィルタリング，すなわち属性に言及しない返信のフィルタリングと定型文のフィルタリングを行う．

3.4.1 属性に言及しない返信のフィルタリング

ユーザがレビュー上で述べている宿泊施設の属性に対する返信を生成するため，レビュー中の属性に言及していない返信を訓練データから除外する．あらかじめ宿泊施設の属性を表す単語 (属性語) の集合 A を定義し，レビューと返信の両方に同じ属性語が出現していない組，ならびに属性語 $a_i \in A$ が 1 つも出現していない組を削除する．これにより，レビューに出現する属性語を含む返信 (属性に言及する返信) が生成される可能性が高まると考えられる．さらに，定型文のみで構成されている返信を訓練データから除外するという効果も期待できる．

属性語は宿泊施設のレビュー集合における特徴的な単語とする．具体的には，式 (3.1) によって各単語のスコアを算出し，その上位 500 件の単語を属性語集合とする．

$$S(w_i) = \mathrm{avg}_{\,r_j \in TOP_{1000}(w_i)}\ \text{TF-IDF}(w_i, r_j) \tag{3.1}$$

ここで，R は苦情ラベルが付与されたレビューの集合，r_j はそのレビュー，TF-IDF(w_i, r_j) は R を全文書集合としたときの単語 w_i のレビュー r_j における TF-IDF，$TOP_{1000}(w_i)$ は TF-IDF 値の大きい上位 1000 件のレビューの集合であり，$S(w_i)$ はその 1000 件の TF-IDF の平均値と定義する．レビュー r_j における単語 w_i の TF-IDF は式 (3.2) で求める．

$$\text{TF-IDF}(w_i, r_j) = tf_{i,j} \times \log\left(\frac{N}{df_i}\right) \tag{3.2}$$

$tf_{i,j}$ は，レビュー r_j における単語 w_i の出現回数，df_i は w_i を含むレビューの数，N は全レビューの数である．$tf_{i,j}$ は TF(Term Frequency)，$\log(\frac{N}{df_i})$ は IDF(Inverse Document Frequency) と呼ばれている．あるレビューにおける TF 値はその単語の出現回数が多いほど高くなるが，そのような単語のうち全てのレビューにおいてよく出現するような単語は IDF 値が低くなる．例えば,「ホテル」という単語は各レビューにおける出現頻度が多いと予想できるため TF 値は高くなるが，全てのレビューにおいてよく出現するため IDF 値は低くなり，結果的に TF-IDF 値は低くなる．本研究では，特定のレビューにおいて高頻度で使用される単語であるが，多くのレビューに出現するような汎用的ではない単語を属性語としている．

属性語抽出の擬似コードを Algorithm 1 に示す．入力はレビューの集合 $R = \{r_1, r_2, \ldots, r_N\}$ とし，出力は属性語集合 A である．$word\_TFIDF$ は，単語をキー，その単語が出現する全てのレビューにおける TF-IDF 値のリストを値とする連想配列である．$word\_avg\_TFIDF$ は，単語をキー，TF-IDF 値の上位 1000 件の平均を値とする連想配列である．

まず，R のそれぞれのレビュー r_j について形態素解析を行い，レビュー内にある単語のリストを取得する (4 行目)．get_words は，文を形態素解析し，単語のリストを取得する関数である．単語リストの各単語 w_i とレビュー r_j を引数として，TF-IDF 値を算出する (6 行目)．単語ごとに，算出した全ての TF-IDF 値を配列として $word\_TFIDF$ に追加する (7 行目-10 行目)．以上の処理を全レビューについて繰り返す (3 行目-12 行目)．次に，$word\_TFIDF$ 内の各単語について，その単語の TF-IDF 値の上位 1000 件を取得し (14 行目)，その平均を計算し，$word\_avg\_TFIDF$ に格納する (15 行目)．最後に，単語を $word\_avg\_TFIDF$ の値の降順にソートし，その上位 500 件の単語を属性語集合 A とする (17 行目)．$sort\_descending\_by\_value$ は，連想配列を値で降順にソートし，そのキー (この場合は単語) を取得する操作である．

Algorithm 1 Extraction of Aspect Words

Input: A set of reviews $R = \{r_1, r_2, \ldots, r_N\}$.
Output: A set of aspect words A.

$word\_TFIDF \leftarrow \{\}$
$word\_avg\_TFIDF \leftarrow \{\}$
for $j = 1$ to N **do**
    $word\_list \leftarrow get\_words(r_j)$
    for w_i in $word\_list$ **do**
        $value \leftarrow$ TF-IDF(w_i, r_j)
        if $w_i \notin word\_TFIDF$ **then**
            $word\_TFIDF[w_i] \leftarrow [\ ]$
        end if
        $word\_TFIDF[w_i].append(value)$
    end for
end for
for all $word, value\_list$ in $word\_TFIDF$ **do**
    $top\_values \leftarrow sort\_descending(value\_list)$[: 1000]
    $word\_avg\_TFIDF[word] \leftarrow average(top\_values)$
end for
$A \leftarrow sort\_descending\_by\_value(word\_avg\_TFIDF)$[: 500]
return A

抽出された属性語の例を表 3.6 に示す．属性語の多くは宿泊施設の属性を表す単語として適切であった．

表 3.6: 宿泊施設の属性語の例

駐車 部屋 排水 予約 タバコ シャワー 臭い 風呂 対応 朝食 タオル 掃除 エアコン 温度 ポイント トイレ 髪の毛 バス 禁煙 喫煙 清掃 空調 ルーム プラン カード 換気 匂い カーテン 料理 冷蔵

3.4.2 定型文のフィルタリング

2.2 節で述べたように，Kew と Volk の研究 [10] では，「訓練データで頻繁に観察される一般的な返信は，モデルに有益でない返信を学習させ，より具体的な返信を生成する能力を低下させる」という仮説に基づき，データセット全体において頻出する表現を除外してから返信生成モデルを学習している．これに倣い，多様な返信を生成するために，すなわち紋切り型の返信が生成されるのを抑制するために，訓練データにおける返信から定型文を除外する．宿泊施設の返信を，句点を境界

として文に分割し，それぞれの文 s_i に対して定型度スコア $C(s_i)$ を算出し，その上位 30%の文を定型文として削除する．この処理の後の訓練データは，レビューと，それに対する元の返信から定型文を除いたテキストの組となる．定型文の削除により返信の全ての文が削除された場合は，その組自体を訓練データから除外する．

定型度スコア $C(s_i)$ は式 (3.3) のように定義する．

$$C(s_i) = \text{ave}_{\,tg_{ij} \in s_i}\, fre(tg_{ij}) \tag{3.3}$$

ここで，tg_{ij} は文 s_i に出現する j 番目の単語 tri-gram，fre は訓練データにおけるその出現頻度であり，定型度スコアはその平均と定義する．訓練データ全体で頻出する単語 tri-gram から構成されている文ほど，その定型度スコアは高くなる．定型度スコアの計算を具体的な例を用いて説明する．表 3.7 は宿泊施設の返信の例であり，表 3.8 はこの返信を句点を境界に文に分割した結果を示している．

表 3.7: 返信の例

ホテルコムズ名古屋をご利用頂きまして誠にありがとうございます。この度は、不愉快な思いをさせてしまい、大変申し訳ございませんでした。頂いた貴重なご意見を肝に銘じて、これから努めてまいります。お忙しい中、ご投稿頂きましてありがとうございました。

表 3.8: 文に分割された返信の例

ID	文
s_1	ホテルコムズ名古屋をご利用頂きまして誠にありがとうございます。
s_2	この度は、不愉快な思いをさせてしまい、大変申し訳ございませんでした。
s_3	頂いた貴重なご意見を肝に銘じて、これから努めてまいります。
s_4	お忙しい中、ご投稿頂きましてありがとうございました。

それぞれの文から単語 tri-gram を抽出する．例えば，文 s_2 からは表 3.9 に示した単語 tri-gram(連続した 3 つの単語の列) が抽出される．なお，アンダーバー (__) を語の境界としている．また，アスタリスク (*) は文頭もしくは文末を表す記号である．

表 3.9: 表 3.8 の s_2 における単語 tri-gram(抜粋)

単語 tri-gram
*__この__度
この__度__は
度__は__、
は__、__不
、__不__愉快
...

訓練データにおける全てのレビューを対象に，文の分割と単語 tri-gram の抽出の処理を行い，それぞれの単語 tri-gram の出現回数をカウントする．その結果，表 3.10 のような統計表が得られる．例えば，訓練データのレビュー全体で,「お__客__様」という単語 tri-gram は 151,701 回出現し,「この__度__は」という単語 tri-gram は 120,338 回出現していることを表す．この表における単語 tri-gram は式 (3.3) における tg_{ij} に相当し，出現頻度は $fre(tg_{ij})$ に相当する．

表 3.10: 単語 tri-gram のレビュー集合全体における出現頻度 (抜粋)

単語 tri-gram	出現頻度
お__客__様	151,701
ござい__ます__*	142,993
まし__た__*	140,660
て__おり__ます	127,033
*__この__度	120,594
この__度__は	120,388
...	...

定型度スコアの算出は表 3.8 に示した単語 tri-gram の出現頻度により算出される．例えば，s_2 の文の定型度スコアは表 3.11 のように計算される．それぞれの単語 tri-gram の訓練データにおける出現頻度の平均値が定型度スコアとなる．

表 3.11: 表 3.8 の文 s_2 の定型度スコアの算出

単語 tri-gram	スコア
*__この__度	120594
この__度__は	120388
度__は__、	33091
は__、__不	1322
、__不__愉快	1595
…	
平均	35241

4 章で後述する評価実験において，本項で述べた定型文のフィルタリングを行った．その結果得られた定型度スコアの分布を図 3.2 に示す．横軸は定型度スコア，縦軸はそのスコアを持つ単語 tri-gram の数である．既に述べたように，本研究では定型度スコアが上位 30%の文を訓練データから取り除く．実験では，上位 30%に該当する定型度スコアの閾値は 30454 となった．定型文のフィルタリングによって除去された返信文の例を表 3.12 に示す．定型度スコアの高い文は，実際に定型的な表現が多いことがわかる．

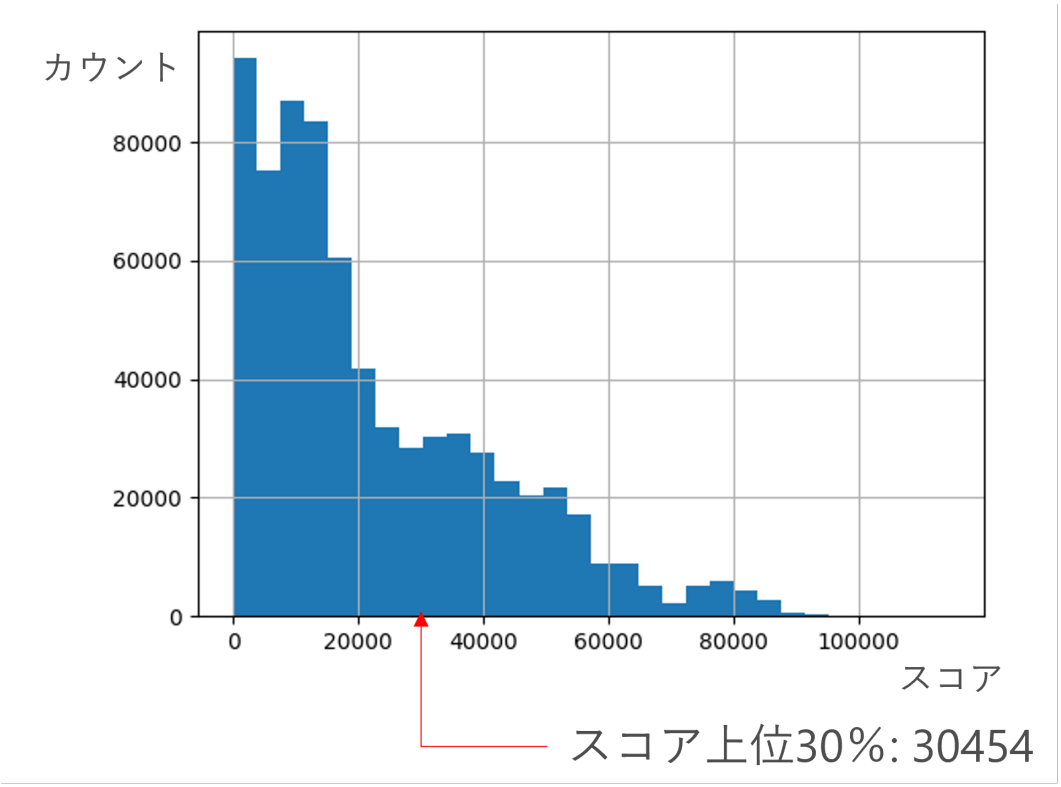

図 3.2: 定型度スコアの分布

表 3.12: 定型度スコアが高い文の例

返信文	スコア
大変申し訳ございませんでした。	78544
心よりお詫び申し上げます。	49978
この度は、ご宿泊頂きまして誠に有難うございます。	48500
またのご来館を心よりお待ち申し上げております。	39078
お客様のご指摘はごもっともと受け止めております。	34997

3.5 返信文の統合

分割したそれぞれのレビュー文から返信生成モデルによって生成された返信文を統合し，最終的な返信を得る．返信文の順序は，生成元のレビュー文のレビューにおける出現順序と同じとする．ただし，返信文は独立に生成しているため，類似した文や表現が重複して返信に含まれる可能性がある．重複する表現を除外するため，2 つの返信文間の距離を正規化された編集距離 [11] で測る．それが 0.1 以下の場合，元のレビューにおける出現順序が後である返信文を残し，もう一方の返信文を除外する．ただし，属性語を含む返信は常に除外しないものとする．

編集距離とは，文字の挿入・削除・置換のいずれかの操作により，ある文字列を別の文字列に変換するために必要な最小操作回数であり，文字列間の類似性もしくは差異を測定するための尺度である．正規化された編集距離とは，編集距離を2つの文字列の長さの和で割った値である．例えば,「こんにちは」という文字列を「こんばんは」という文字列にするには，3文字目の「に」を「ば」に置換し，4文字目の「ち」を「ん」に置換すればよいので，編集距離は2となる．正規化された編集距離は，2つの文字列の長さが共に5であるため，$\frac{2}{5+5} = 0.2$となる．

返信の統合の例を示す．表3.13は表3.2の再掲である．レビューを文s_1と文s_2に分割し，それぞれの文から返信を生成している．ここでは，文s_1も文s_2も，それに対して生成された返信は3つの文から構成される．なお，s_{mn}はm番目のレビュー文に対して出力されたn番目の返信文を表す．

表 3.13: 文分割と返信の自動生成の例 (再掲)

文に分割されたレビュー	返信
s_1 駅から近いが、お風呂の水はけが悪かった。	s_{11} お風呂の水はけが悪かったとのことで、大変申し訳ございませんでした。 s_{12} 今後このようなことがないよう、清掃スタッフに指導して参ります。 s_{13} またのご利用をお待ち申し上げております。
s_2 またお風呂の換気扇の音が電気を消してから相当たってから大きな音で「ビューン」といって切れるので、睡眠の妨げになった。	s_{21} お風呂の換気扇の音につきましては、大変申し訳ございませんでした。 s_{22} 今後このようなことのないよう、点検を徹底して参ります。 s_{23} またのご利用を心よりお待ち申し上げております。

生成された全ての返信文の組について，その正規化編集距離を計算する．表3.14はその計算結果を三角行列として示したものである．

表 3.14: 返信文同士の正規化編集距離

	s_{11}	s_{12}	s_{13}	s_{21}	s_{22}	s_{23}
s_{11}	0					
s_{12}	0.78	0				
s_{13}	0.77	0.73	0			
s_{21}	0.35	0.81	0.81	0		
s_{22}	0.77	0.28	0.66	0.80	0	
s_{23}	0.79	0.70	0.07	0.82	0.68	0

s_{13} と s_{23} に着目すると，その編集距離は，s_{23} から「心より」という長さ 3 の文字列を削除すると s_{13} と一致するため，3 となる．s_{13} と s_{23} の文字長はそれぞれ 20, 23 なので，正規化編集距離は $\frac{3}{20+23} \approx 0.07$ となる．この値はあらかじめ設定した閾値 0.1 以下であるため，このうち出現順序が後である s_{23} を残し，s_{13} を削除する．一方，正規化編集距離 0.1 以下の返信文の組は s_{13} と s_{23} の組以外には存在しない．その結果，表 3.15(a) に挙げた 5 つの返信文が残される．最後に，これらの返信文を統合し，表 3.15(b) に示す返信が出力として得られる．

表 3.15: 返信の統合の例

(a) 類似文を除いた後の文

s_{11}	お風呂の水はけが悪かったとのことで、大変申し訳ございませんでした。
s_{12}	今後このようなことがないよう、清掃スタッフに指導して参ります。
s_{21}	お風呂の換気扇の音につきましては、大変申し訳ございませんでした。
s_{22}	今後このようなことのないよう、点検を徹底して参ります。
s_{23}	またのご利用を心よりお待ち申し上げております。

(b) 最終的な返信

お風呂の水はけが悪かったとのことで、大変申し訳ございませんでした。今後このようなことがないよう、清掃スタッフに指導して参ります。お風呂の換気扇の音につきましては、大変申し訳ございませんでした。今後このようなことのないよう、点検を徹底して参ります。またのご利用を心よりお待ち申し上げております。

第4章　評価

本章では，3 章で述べた提案手法の評価を行う．4.1 節では評価実験に利用したデータセットについて述べる．4.2 節では苦情判定モデルを評価する．4.3 節では，返信文生成モデルにより生成された返信を評価する．4.4 節では，提案手法によって生成された返信の例を確認し，それに対する考察を行う．4.5 節では，近年急速に発展を遂げている大規模言語モデルである ChatGPT[4] により生成した返信を評価する．

4.1　データセット

実験には楽天トラベルデータセット [14] を利用する．楽天トラベルデータセットには，楽天トラベルのユーザが利用した宿泊施設に関してのレビューと，それに関する様々な情報 (項目) が記載されている．これらの項目のうち，本研究では以下の 3 項目を利用する．

- **ユーザ投稿本文**: ユーザが宿泊施設に対して投稿したレビュー
- **分類**: レビューの内容に関する分類カテゴリ．「感想・情報」「苦情」などの文字列がある．
- **施設回答本文**: ユーザが投稿したレビューに対する宿泊施設の返信．返信がない場合もある．

楽天トラベルデータにはのべ 5,582,428 件のレビューが含まれる．本実験ではその一部を使用する．実験に使用するデータ数の詳細は 4.2.1 項と 4.3.1 項で述べる．

4.2　苦情判定モデルの評価

本節では，3.3 節で説明した苦情判定を評価する．4.2.1 項では実験条件を述べ，4.2.2 項では実験結果の報告及び考察を行う．4.2.3 項では，モデル学習時のハイパーパラメタを変更した追加実験について述べる．

4.2.1 実験条件

苦情判定モデルの学習ならびに評価に用いたデータの統計を表 4.1 に示す．3.3 項で述べたように，楽天トラベルデータから，1 文から構成される (句点が 1 つだけある) レビューのうち，分類の項目に「苦情」ラベルが付与されているものを正例 (苦情のレビュー) として取得する．次に，1 文から構成されかつ「感想・情報」ラベルが付与されているレビューの中から正例と同じ数のレビューを負例 (苦情ではないレビュー) として取得する．このように獲得したデータを分割し，その 80% を訓練データ，20%をテストデータとする．

表 4.1: 苦情判定モデルの実験データ

	正例	負例	合計
訓練データ	16,099	16,099	32,198
テストデータ	4,025	4,025	8,050
合計	20,124	20,124	40,248

BERT をファインチューニングする際のハイパーパラメタとして，学習率は $2e^{-5}$，その他は transformers.AdamW [3] のデフォルト値とした．

4.2.2 実験結果・考察

苦情判定モデルの評価基準として，二値分類の正解率，および苦情クラス (正例) の精度，再現率，F 値を用いる．ここで，正解率，精度，再現率，F 値は，表 4.2 に示す混同行列を元に，式 (4.1)，(4.2)，(4.3)，(4.4) のように定義される．なお，本実験では，Positive は苦情のレビュー，Negative は苦情ではないレビューを表す．

表 4.2: 一般的な二値分類の混同行列

		モデルの予測	
		Positive	Negative
実際のクラス	Positive	True Positive(TP)	False Negative(FN)
	Negative	False Positive(FP)	True Negative(TN)

$$\text{正解率} = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.1}$$

$$\text{精度} = \frac{TP}{TP + FP} \tag{4.2}$$

$$\text{再現率} = \frac{TP}{TP + FN} \tag{4.3}$$

$$\text{F 値} = 2 \cdot \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4.4}$$

表 4.3 に苦情判定モデルの評価結果を示す.「epoch 数」は BERT をファインチューニングする際の epoch 数を表す．正解率や F 値は 0.9 に近く，苦情判定モデルの性能が十分に高いことが確認された．また，epoch 数が 1 のときに正解率や F 値が最大となり，epoch 数を増加させると全体的な性能が低下する傾向が見られた．この原因として，学習が進むにつれて過学習を起こしている可能性が考えられる．以後の実験では epoch 数が 1 の苦情判定モデルを用いる．

表 4.3: 苦情判定モデルの評価

epoch 数	正解率	精度	再現率	F 値
1	**0.8877**	0.8718	0.9091	**0.8901**
3	0.8847	0.8656	**0.9108**	0.8877
5	0.8744	**0.8768**	0.8713	0.8740
7	0.8758	0.8587	0.8996	0.8787
9	0.8694	0.8656	0.8748	0.8701

4.2.3 追加実験

一般に，BERT では epoch 数が進むにつれて性能が良くなり，epoch 数が十分に大きくなると性能の改善が収束するか，過学習により性能が低下する．一方，4.2.2 項で示した実験結果では，本研究の苦情判定モデルは epoch 数が 1 の時にもっとも高い性能を示した．epoch 数と苦情判定モデルの性能の関係を更に分析するため，ハイパーパラメータを変更して epoch 数を変化させたときの苦情判定モデルの性能を評価する追加実験を行う．4.2.2 項では，学習率を $2e^{-5}$ としていたが，追加実験ではこれを $5e^{-6}$，$1e^{-6}$ に変更して実験する．それぞれの実験結果を表 4.4，4.5 に示す.

正解率と F 値は，どちらの学習率でも epoch 数 1 のときに最大となった．再現率に関しても，epoch 数が 1 のとき，学習率が $1e^{-6}$ では最高となり，$5e^{-6}$ のときにも最大の再現率 (epoch 数 9 のとき) とほぼ同じであった．前項での実験も含め，いずれの学習率においても，epoch 数が 1 の時に正解率が一番良いという傾向が確認された．異なる学習率で学習された苦情判定モデルの正解率や F 値を比較すると，正解率は学習率が $5e^{-6}$ のとき，F 値は学習率が $1e^{-6}$ のときに，一番高い値が得られた．しかし，これらに大きな差はなかった．したがって，既に述べたように，以後の実験では学習率を $2e^{-6}$ と設定して学習したモデル (表 4.3 のモデル) を用いる．

表 4.4: 苦情判定モデルの評価 (学習率 $5e^{-6}$)

epoch 数	正解率	精度	再現率	F 値
1	**0.8949**	0.8809	0.9133	**0.8968**
3	0.8776	**0.8960**	0.8544	0.8747
5	0.8719	0.8771	0.8651	0.8710
7	0.8755	0.8722	0.8800	0.8761
9	0.8824	0.8581	**0.9163**	0.8862

表 4.5: 苦情判定モデルの評価 (学習率 $1e^{-6}$)

epoch 数	正解率	精度	再現率	F 値
1	**0.8842**	0.8659	**0.9093**	**0.8871**
3	0.8810	0.8721	0.8929	0.8824
5	0.8829	**0.8728**	0.8964	0.8844
7	0.8825	0.8705	0.8986	0.8844
9	0.8810	0.8633	0.9053	0.8838

4.3 返信生成モデルの評価

本節では，3 章で述べた方法によって生成した返信に対して，自動評価と人手評価の両方を行う．4.3.1 項では実験条件を述べる．4.3.2 項では，本実験で比較する手法について説明する．4.3.3 項では，自動評価の手続きを述べ，実験結果を報告し，考察する．4.3.4 項では，人手評価の手続き，結果，考察について述べる．

4.3.1 実験条件

返信生成モデルの学習ならびに評価に用いたデータの統計を表 4.6 に示す．実験データは，楽天トラベルデータにおいて，レビューの分類が「苦情」であり，かつ施設回答本文が存在する (空でない) レビューを用いた．その 90%を訓練データ，5%を開発データ，5%をテストデータとした．開発データは研究の初期段階で訓練データのフィルタリング手法を検討する際に，出力を確認するために用いた．3.4.1 項と 3.4.2 項で述べたフィルタリング処理により，訓練データのデータ数は約 29% 減少した．

返信生成モデルとして使用した BART をファインチューニングする際のハイパーパラメタとして，最大エポック数は 5，学習率は $3e^{-5}$，ドロップアウト率は $p = 0.3$ と設定した．

表 4.6: 返信生成モデルの実験データ

訓練データ	147,749
訓練データ (フィルタリング後)	105,241
開発データ	8,209
テストデータ	8,209

4.3.2 比較手法

本実験では，以下に述べる 6 つの手法を比較する．また，これらの手法によって生成された返信に加え，実際に宿泊施設が書いた返信も比較する．

BASELINE BART モデルによってレビューから返信を生成するモデル．訓練データに対するフィルタリングは行わない．レビューを文に分割してから返信を生成するのではなく，レビュー全体を入力として返信を生成する．これをベースラインとする．

BASELINE-S もう 1 つのベースライン手法．訓練データに対するフィルタリングは行わない．図 3.1 に示したようにレビューを文に分割しそれぞれの文から生成された返信を統合する処理は行う．

PRO-A-S 訓練データに対して属性に言及しない返信のフィルタリング (3.4.1 項) を行い，返信生成モデルを学習する手法．文分割と返信文の統合処理も行う．

PRO-C-S 訓練データに対して定型文のフィルタリング (3.4.2 項) を行い，返信生成モデルを学習する手法．文分割と返信文の統合処理も行う．

PRO-AC レビューを文に分割してから返信を生成するのではなく，レビュー全体を入力として返信を生成する手法．訓練データに対する上記 2 つのフィルタリング処理も行う．

PRO-AC-S 訓練データに対して 2 つのフィルタリング処理を行い，文分割と返信文の統合処理も行う手法．

GOLD データセットにおいて宿泊施設が実際に書いた返信

比較手法の特徴を表 4.7 にまとめる．「フィルタリング」における「属性」の列は，属性に言及しない返信のフィルタリングを適用するかしないかを示す．適用するとき，PRO-A-S のように，手法の略号に「A」をつける．「フィルタリング」における「定型文」の列は，定型文を除外するフィルタリングを適用するかしないかを表す．適用するとき，手法の略号に「C」をつける．「文分割」の列は，レビューを文に分割し，文毎に返信を生成し，最後にそれらを統合する処理を行うか，この処理を行わずにレビュー全体を入力として返信を生成するかを示す．文毎に返

信を生成するとき，手法の略号に「S」をつける．また，手法の略号が「PRO」で始まるものは提案手法を，それ以外はベースラインを表す．

表 4.7: 比較手法のまとめ

	フィルタリング		文分割 (S)
	属性 (A)	定型文 (C)	
BASELINE	×	×	×
BASELINE-S	×	×	✓
PRO-A-S	✓	×	✓
PRO-C-S	×	✓	✓
PRO-AC	✓	✓	✓
PRO-AC-S	✓	✓	×

自動評価では，GOLD 以外の 6 つの手法を評価する．一方，人手評価では，BASELINE を除く 5 つの手法を評価する．さらに，返信生成タスクの上限として，GOLD の返信も人手によって評価する．

4.3.3 自動評価

評価指標

ここでは訓練データのフィルタリングの効果を自動的に評価する．自動評価尺度として BLEU[15] と DISTINCT[13] を用いる．

BLEU は，モデルが出力したテキスト (candidate) が人間が書いたテキスト (reference) とどれだけ類似しているかを測定する指標である．このスコアは 0 から 1 までの値をとり，1 に近いほどモデルの出力が人間が書いたテキストと類似していることを意味する．

基本的には，candiate と reference とで単語 n-gram がどれだけ重複しているかで両者の類似度を測る．BLEU スコアの計算式は式 (4.5)，(4.6)，(4.7) の通りである．ただし，r，c はそれぞれ reference，candidate の文字長，N は比較する n-gram の最大の長さを表す．また，$w_n = \frac{1}{N}$ である．

$$\mathrm{BLEU} = \mathrm{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{4.5}$$

$$\mathrm{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{\frac{1-r}{c}} & \text{otherwise} \end{cases} \tag{4.6}$$

$$p_n = \frac{\sum_{C \in \{\mathrm{Candidates}\}} \sum_{\text{n-gram} \in C} \mathrm{Count}_{\mathrm{clip}}(\text{n-gram})}{\sum_{C' \in \{\mathrm{Candidates}\}} \sum_{\text{n-gram}' \in C'} \mathrm{Count}(\text{n-gram}')} \tag{4.7}$$

ここで，$Count_{clip}$ は，

$$Count_{clip}(\text{n-gram}) = min(Count(\text{n-gram}), MaxRefCount(\text{n-gram}))$$

と定義される．$Count_{clip}$ は，n-gram の candidate における出現回数であるが，その回数は reference での n-gram の最大出現数を超えないように制限されている．なお，$MaxRefCount$ は n-gram が reference 文中に出現する回数を表している．また，BP，p_n の役割はそれぞれ以下の通りとなる．

BP ：candidate が短すぎる場合にペナルティを適用し，過剰に短いテキストに高いスコアを与えないようにする．

p_n ：candidate の各 n-gram が reference にどれだけ存在するかを計算する．

本研究では，reference をデータセット内の実際の宿泊施設の返信，candidate をモデルにより生成した返信とし，BLEU を算出する．

DISTINCT は，評価データに対して生成された返信テキストの多様性を評価する指標である．具体的には生成されたテキストにおけるユニークな N-gram の多様性を測る．その計算式を式 (4.8) に示す．

$$\text{distinct-N} = \frac{\text{Number of distinct N-grams in } T}{\text{Total number of N-grams in } T} \tag{4.8}$$

ここで，分子はテキスト T における異なる N-gram の数，分母はテキスト T における N-gram の総数である．この比率は 0 から 1 の間の値を取り，1 に近いほどテキストに多様性があることを示す．本研究では，テキスト T をモデルにより生成された返信全体とし，DISTINCT を算出する．

BLEU も DISTINCT も，n-gram の n を自由に設定できる．本実験では，単語 4-gram を基にした指標 (BLEU-4 と DISTINCT-4) を用いる．

結果と考察

表 4.8 に 6 つの手法による返信の BLUE-4 と DISTINCT-4 を示す．

フィルタリング処理の効果について検証する．BASELINE と PRO-AC，BASELINE-S と PRO-AC-S をそれぞれ比較すると，フィルタリング処理を行う PRO-AC，PRO-AC-S の方が DISTINCT-4 が高い．このことから，定型文のフィルタリングにより，ありきたりな文の生成が抑制され，様々な表現の文が生成されるようになったと言える．一方，BLEU-4 については，フィルタリング処理をしない BASELINE，BASELINE-S が高い．これは，フィルタリング処理をしないデータから学習されたモデルでは定型文が生成されることが多いが，評価データにおける正解の返信にも定型文が多く，両者で単語 n-gram が一致することが多いためと考えられる．

次に，文毎に返信を生成する処理の効果について考察する．BASELINE と BASELINE-S，PRO-AC と PRO-AC-S を比較すると，DISTINCT-4 については前者の方が高い．すなわち，文毎に返信を生成することによって DISTINCT-4 が低下している．これは，各生成文を統合する際に n-gram の総数は増えるが，統合した結果の各文には類似した表現が多いため，ユニークな n-gram の総数はあまり増加しないためと考えられる．なお，文毎に返信し，それらを統合して最終的な返信を生成する例については，4.4 項で詳しく述べる．

最後に 2 つのフィルタリング処理の違いについて考察する．ここでは PRO-A-S(属性に言及しない返信のフィルタリングを使用)，PRO-C-S(定型文のフィルタリングを使用)，PRO-AC-S(両方のフィルタリングを使用) を比較する．BLEU-4 に関しては，PRO-A-S，PRO-C-S，PRO-AC-S の順に高い値を示した．このことは，GOLD に近い返信を生成するという観点においては，定型文のフィルタリングに比べ，属性に言及しないフィルタリングの方が効果的であることを示唆している．DISTINCT-4 については，PRO-A-S，PRO-AC-S，PRO-C-S の順に高い結果が得られた．この結果から，多様な表現を生成するという観点でも，属性に言及しない返信のフィルタリングの方が定型文のフィルタリングより効果的であることが分かる．属性に言及していない文を訓練データから除外することで，結果として定型文もあわせて除外されたのではないかと推察できる．

表 4.8: 返信生成モデルの自動評価結果

	BLEU-4	DISTINCT-4
BASELINE	**0.1233**	0.0313
BASELINE-S	0.1034	0.0224
PRO-A-S	0.0962	0.0562
PRO-C-S	0.0740	0.0395
PRO-AC	0.0660	**0.0585**
PRO-AC-S	0.0667	0.0533

4.3.4 人手評価

実験手順

表 4.6 に示したテストデータからランダムに 50 件のレビューを選択し，4.3.2 項に示した 5 つの手法で生成された返信ならびに GOLD を人手で評価する．評価者は日本語母語話者 7 名である．評価項目と，実際に被験者に提示した評価基準を以下に記載する．

流暢性 返信が自然な日本語であるかを 5 段階で評価する．評価基準は以下の通り．

1. 日本語として文法的な誤りがかなり多い

2. 日本語として文法的誤りを多少含む
3. 文法的誤りはないが多少不自然さがある
4. 日本語として概ね自然である
5. 日本語として非常に自然である

非冗長性 返信に同じような表現が繰り返されていないかを5段階で評価する．表現の繰り返しが多いほど低い評点を与える．評価基準は以下の通り．

1. ほぼ同じ文の繰り返しが何回もある
2. ほぼ同じ文の繰り返しがある
3. 内容は違うがほぼ同じ言い回しの繰り返しが何回もある
4. 内容は違うがほぼ同じ言い回しの繰り返しがある
5. 同じ内容・言い回しが繰り返されることはほとんどない

総合評価 苦情を書いたレビュワーの立場から見て，宿泊施設からの返信として適切であるかどうかを5段階で評価する．評価基準は以下の通り．

1. 非常に不適切である
2. 不適切である
3. どちらとも言えない
4. 適切である
5. 非常に適切である

属性への言及 レビューでユーザが苦情を述べている属性のそれぞれについて，返信でそれについて触れているか否かを判定する．属性はあらかじめ被験者以外の人が抽出しておく．

被験者には，元のレビューと，それに対して異なるシステムで生成された返信が提示される．この際，どの返信がどのシステムで生成されたかわからないようにし，かつ表示する6種類の返信の順番もランダムに変更する．

人手による評価の例を図4.1に示す．元のレビューは「トイレを暖房付きで、シャワートイレを望みます。 テレビの映りが悪いのはどうにかなりませんか。」という内容で，あらかじめ「トイレの設備」「テレビの映り」という2つの不満の属性が被験者以外の作業者により抽出されている．被験者は，6つの返信の自然さ，非冗長性，総合評価を5段階で評価し，またそれぞれの不満の属性が返信内で言及されているか否かを記入する．

レビュー	返信	自然さ	冗長性	総合評価	属性1	属性2
トイレを暖房付きで、シャワートイレを望みます。 テレビの映りが悪いのはどうにかなりませんか。					トイレの設備	テレビの映り
	テレビの映りが悪いとのことで、大変申し訳ございませんでした。今後はこのようなことがないよう、点検を徹底して参ります。	5	5	3	なし	あり
	またのご利用を心よりお待ち申し上げております。	5	5	1	なし	なし
	テレビの映りが悪いとのことで、大変申し訳ございませんでした。今後はこのようなことのないよう、スタッフ一同努めて参ります。	5	5	2	なし	あり
	この度はご利用いただきまして誠にありがとうございます。シャワートイレの設置につきましては本館全ての客室に設置を完了しております。残念ながら暖房付ではございませんが今後につきましては検討中でございます。また、テレビ映像の不具合につきまして誠に申し訳ございませんでした。電波状況及びセレクター、チューナーの不具合が発覚し、現在調整しております。お客様にはかさねてお詫び申しあげるとともに次回のご利用をお待ち申し上げております。 伊良湖ガーデンホテル　支配人	5	5	5	あり	あり
	テレビの映りが悪いとのことで、大変申し訳ございませんでした。今後はこのようなことがないよう、点検を徹底して参ります。またのご利用を心よりお待ち申し上げております。	5	5	3	なし	あり
	テレビの映りが悪いとのことで、大変申し訳ございませんでした。今後はこのようなことがないよう、スタッフ一同精進して参ります。	5	5	2	なし	あり

図 4.1: 人手評価の例

結果と考察

人手評価による実験結果を表 4.9 に示す．評価値は 7 名の被験者による評点の平均である．アスタリスク (*) は t 検定によって BASELINE との有意差 ($p < 0.01$) があることを示す．被験者による判定の一致度を測るため，全ての被験者の組について Fleiss の κ 係数 [7] を測り，その平均値を求めた．その結果，流暢性の評点の平均 κ 係数は 0.34，非冗長性は 0.62，属性言及数は 0.77，総合評価は 0.42 となり，7 名の被験者による評価はある程度一貫していることがわかった．

表 4.9: 返信生成モデルの人手評価の結果

	流暢性	非冗長性	総合評価	属性言及率
BASELINE-S	4.58	4.44	2.53	0.338
PRO-A-S	4.34*	4.16*	2.72*	0.581*
PRO-C-S	4.63	4.50	2.80*	0.415*
PRO-AC	4.66	4.71*	2.98*	0.450*
PRO-AC-S	4.48	4.17*	2.77*	0.538*
GOLD	4.63	4.84	3.99	0.652

流暢性に着目すると，異なるモデル間で流暢性のスコアの差はほとんどなく，また GOLD との差もそれほど大きくない．したがって，全てのモデルがある程度自然な文を生成できている．

属性に言及しない返信のフィルタリングに着目すると，BASELINE-S と比べて，属性に言及しない返信のフィルタリングを行う提案手法 (PRO-A-S，PRO-AC-S) では，属性言及率が高い．ユーザが苦情を述べている属性に対して何らかの返信をするという本研究の目的がある程度達成できている．一方で，流暢性と非冗長性のスコアは BASELINE-S と比べて低くなっている．これは，属性に対する言及が増えたことにより，同じような表現の繰り返しが増え，流暢性も損われたと考えられる．提案手法では，個々のレビュー文に対する返信文を統合する際に類似した返信文を除外する処理をしているが，属性を含む文は除外しないことにしているため，似ている表現を完全に排除できていない．属性言及率と流暢性・非冗長性はトレードオフの関係にあると言える．

定型文のフィルタリングに着目すると，PRO-C-S は BASELINE-S と比べて非冗長性が改善され，流暢性や総合評価も高く，紋切り型の表現の生成が抑制されていることが確認できた．ただし，定型文のフィルタリングのみを行う PRO-C-S の手法は，属性に言及しない返信のフィルタリングを行う PRO-A-S や PRO-AC-S に比べて属性言及率が低い．属性言及率を上げるという観点から見ると，定型文のフィルタリングは属性に言及しない返信のフィルタリングよりも効果が低いことがわかった．

文毎に返信を生成しそれを統合する処理の有効性について検証する．PRO-AC と PRO-AC-S を比較すると，属性言及率は PRO-AC-S の方が高いが，流暢性・非冗長性の指標は PRO-AC の方が高い．文ごとに返信を生成し，最終的にそれを統合する方法では，レビュー中の属性に対して漏れなく言及することができるが，レビュー全体を一括して処理する手法と比べて文の自然さが損われたり類似表現の繰り返しが生じているためである．総合評価では PRO-AC の方が高いことから，文ごとに返信を生成する方式の有効性は認められない．ただし，文毎に返信を生成する手法は，レビューの中に複数の属性があるとき，その一部の属性が返信で言及されないことを避けることを目的としている．そのため，属性が複数あるレビューについて提案手法が有効に働くかを検証する必要がある．

ユーザが不満を表明した属性が 1 つしかないレビューと 2 つ以上あるレビューのみを対象に提案手法を評価する．属性が 1 つしかない 28 件のレビューのみを対象にした評価結果を表 4.10 に，属性が 2 つ以上存在する 22 件のレビューのみを対象にした評価結果を表 4.11 に示す．属性が 1 つだけのレビューのみを評価の対象としたとき，レビューを文に分割しない PRO-AC の属性言及率は BASELINE-S に比べ大幅に高く，総合評価も文ごとに返信を生成するモデルに比べ高い．一方，属性が複数存在するレビューのみを評価の対象としたときは，PRO-AC の属性言及率は BASELINE-S よりも悪く，総合評価も PRO-AC-S と比べて低い．

以上から，属性が 1 つしかないレビューに対しては，文ごとに返信を生成しても属性言及率を上げる効果は小さいが，複数の属性を含むレビューについては，レビューを文に分割してから返信を生成する提案手法が有効に働くと言える．また，表 4.11 における属性言及率・総合評価の両方について，PRO-AC と比べて他の提

案手法が高くなっていることから，複数の属性を含むレビューを評価対象にしたときは，レビュー内で言及されている全ての不満に対して返信することは総合評価を向上させることにつながると考えられる．

表 4.10: 属性が 1 つだけのレビューに対する返信生成モデルの人手評価の結果

	流暢性	非冗長性	総合評価	属性言及率
BASELINE-S	4.61	4.57	2.57	0.371
PRO-A-S	*4.39	*4.29	*2.83	*0.665
PRO-C-S	4.71	4.65	2.79	*0.482
PRO-AC	4.70	4.70	*3.27	*0.619
PRO-AC-S	4.47	*4.33	*2.81	*0.594
Gold	4.70	4.85	4.04	0.70

表 4.11: 複数の属性を含むレビューに対する返信生成モデルの人手評価の結果

	流暢性	非冗長性	総合評価	属性言及率
BASELINE-S	4.54	4.27	2.48	0.296
PRO-A-S	4.27*	3.99*	2.58	0.473*
PRO-C-S	4.54	4.30	2.81*	0.329*
PRO-AC	4.60	4.73*	2.61	0.232
PRO-AC-S	4.50	3.97*	2.73*	0.466*
GOLD	4.56	4.82	3.94	0.596

4.4 返信の生成例

本節では，実際に出力された返信を例に，提案手法の有効性や問題点を述べる．あるレビューに対する BASELINE-S，PRO-AC-S, GOLD の返信を表 4.12 に示す．このレビューでユーザが不満を表しているのは下線部の「禁煙室の案内」「風呂の温度」「シャワーの強さ」の 3 点である．BASELINE では「風呂の温度」に対してしか謝罪していないのに対し，PRO-AC-S では 3 つの属性全てに言及して謝罪している．

表 4.12: 生成された返信の例

レビュー	返信		
	BASELINE	PRO-AC-S	GOLD
まず最初に、全室禁煙なら大々的に謳って欲しい。サイトのどこかには記載があるだろうけど、ざっくり読む分には見つけきらない。 係りの方は雰囲気の良い方で、満足です。客室のデザインは良いが、お風呂がない。 貸切風呂（一番左奥）は温度調整できず真夏に熱い風呂と罰ゲームでした。 シャワーが弱すぎる。最終的には湯船の熱いお湯を汲み利用しました。 大浴場は普通で良かった。 食事は素材の味を生かし、的な内容でした。悪く言えばそんなに手をかけていない。 っと思いきや、唯一一品だけ、大根のレモン煮 (?) 抜群に美味かった。これだけは、感動しました。	貸切風呂の温度につきましては、ご迷惑をお掛け致しまして申し訳ございませんでした。お客様に快適にお過ごし頂けるよう、スタッフ一同努力して参ります。この度は当ホテルをご利用頂きまして誠にありがとうございます。	禁煙ルームのご案内につきましては、ご不便をお掛け致しました事、心よりお詫び申し上げます。ご指摘いただきましたサイトの件でございますが、お客様にご迷惑をお掛け致しましたことを深くお詫び申し上げます。貸切風呂の温度調整につきましては、お客様にご迷惑をお掛け致しました事を心よりお詫び申し上げます。シャワーの件では大変申し訳ございませんでした。今後このような事がないよう、スタッフ一同精進して参ります。	湯の宿　入船でございます。 先日はご宿泊頂きまして、誠にありがとうございました。 ご指摘頂きました箇所に関しましては、出来る箇所から順次、改善してまいりたいと思います。 今後とも、入船を宜しくお願い致します。

表 4.12 に示したレビューを文に分割し，苦情判定モデルで判定した結果と，それぞれのレビュー文を入力として PRO-AC-S の返信生成モデルで返信文を生成した結果を表 4.13 に示す．苦情判定の列において，0 は苦情と判定されていない文，1 は苦情と判定された文であることを表す．なお，提案手法では苦情と判定されなかった文に対して返信を生成しないが，表 4.13 では参考のためモデルによって生成された返信を表示している．

苦情判定に着目すると，ある程度苦情判定モデルが有効に働いていることがわかる．例えば，s_8 の文は「大浴場は良かった」であり，苦情ではないにも関わらず，返信生成モデルは「大浴場」に関して謝罪している．苦情判定の処理をせずに全ての文に対して返信を生成し，そのまま統合処理を行うと，本来謝罪をする必要のない属性に対して謝罪をしてしまうことになる．しかし，苦情判定モデルはこの文を苦情ではないと判定しているため,「大浴場」に対する謝罪を表す文は実際には生成されない．一方で，s_4 のように苦情とも取れるような文を苦情では

ないと判定したり，s_{10} のように苦情とはいえない文に対して苦情と判定したりと，改善の余地がある．

苦情と判定された文ごとに返信文を生成するという手法に関し，複数の属性に言及するという目的がある程度達成されていることがわかる．前述した通り，このレビューでは「禁煙室の案内」「風呂の温度」「シャワーの強さ」という 3 つの点で不満が述べられており，それらは文としてはそれぞれ s_1，s_5，s_6 に表明されている．そして，それぞれの文に対して返信を生成することより，最終的にこれらの属性の全てに対して言及した返信が得られている．一方，1 つの属性に対する不満が複数の文によって表されていることがある．提案手法は，属性は 1 つの文で表されていることを仮定しているため，このような場合に冗長な返信を生成する可能性がある．例えば,「禁煙室の案内」という属性に対する不満は文 s_1 と文 s_2 で述べられている．文 s_1 に対する返信で「禁煙室の案内」に，文 s_2 に対する返信で「サイト」に言及し，これらを統合して最終的な返信を生成しているが,「禁煙室の案内」と「サイト」の両方に言及するのは冗長であるとも言える．レビューを句点で分割するのではなく，属性を 1 つだけ含むパッセージに分割することで，より適切にユーザの不満に答える返信を生成できると考えられる．

返信生成モデルが生成する文に目を向けると，s_1 のように「禁煙ルームの案内」と苦情の内容をうまく言い換えられている返信がある一方で，s_2 の「サイト」や s_6 の「シャワー」のようにレビュー内の単語を拾って，それについて謝罪をするという場合も多い．生成される文もあまりバリエーションがなく,「〜の件では」や「〜につきましては」などの定型的な表現とも言える文が散見された．また，不満に対する改善策を述べるということはあまり実現できてない．現状の返信生成モデルには以上のような課題がある．

返信文の統合処理に関しては，s_4 の「またのご利用を心よりお待ち申し上げております」と s_6 の「またのお越しを心よりお待ち申し上げております」といった似ている文があったとき，s_4 が削除されている．一方で，3.5 節でも述べた通り，属性を含む文は除外しないことにしているため，類似した表現を完全には排除できていない．統合処理によって得られる返信では「お詫び申し上げます」で終わる文が続き，全体として冗長になってしまっている．

表 4.13: 文毎に返信を生成する処理の例

番号	分割したレビュー	苦情判定	生成された文
s_1	まず最初に、全室禁煙なら大々的に謳って欲しい	1	禁煙ルームのご案内につきましては、ご不便をお掛け致しました事、心よりお詫び申し上げます。
s_2	サイトのどこかには記載があるだろうけど、ざっくり読む分には見つけきらない	1	ご指摘いただきましたサイトの件でございますが、お客様にご迷惑をお掛け致しましたことを深くお詫び申し上げます。今後このようなことのないよう、スタッフ一同サービスの向上に努めて参ります。
s_3	係りの方は雰囲気の良い方で、満足です	0	お客様には大変ご不快な思いをさせてしまいました事、心よりお詫び申し上げます。
s_4	客室のデザインは良いが、お風呂がない	0	お風呂の件ではご迷惑をお掛け致しまして申し訳ございませんでした。今後このような事がないよう、スタッフ一同努力して参ります。またのご利用を心よりお待ち申し上げております
s_5	貸切風呂（一番左奥）は温度調整できず真夏に熱い風呂と罰ゲームでした	1	貸切風呂の温度調整につきましては、お客様にご迷惑をお掛け致しました事を心よりお詫び申し上げます。
s_6	シャワーが弱すぎる	1	シャワーの件では大変申し訳ございませんでした。今後このような事がないよう、清掃スタッフに指導して参ります。またのお越しを心よりお待ち申し上げております
s_7	最終的には湯船の熱いお湯を汲み利用しました	0	お客様には大変ご迷惑をお掛け致しました事を心よりお詫び申し上げます。
s_8	大浴場は普通で良かった	0	大浴場につきましては、ご不便をお掛け致しました事、心よりお詫び申し上げます。
s_9	食事は素材の味を生かし、的な内容でした	0	お食事の内容につきましては、お客様にご不快な思いをさせてしまい誠に申し訳ございませんでした。今後このような事がないよう、スタッフ一同精進して参ります。またのお越しを心よりお待ち申し上げております
s_{10}	悪く言えばそんなに手をかけていない	1	お客様には大変不愉快な思いをさせてしまいました事、心よりお詫び申し上げます。今後このような事がないよう、スタッフ一同精進して参ります。
s_{11}	っと思いきや、唯一一品だけ、大根のレモン煮（？）抜群に美味かった	0	せっかくのご旅行を台無しにしてしまい誠に申し訳ございませんでした。
s_{12}	これだけは、感動しました	0	せっかくのご旅行に不快な思いをさせてしまい誠に申し訳ございませんでした。心よりお詫び申し上げます。

4.5 ChatGPTによる返信生成の考察

大規模言語モデル (Large Language Model; LLM) の中には，特定の下流タスクのための訓練データを必要とせず，適切なプロンプトを用いることで，様々な自然言語処理タスクに対応できるものが存在する．近年，そのようなLLMの1つで，OpenAIが開発したChatGPT[4]が注目を集めている．ChatGPTは，GPT-3.5及びGPT-4系のモデルを基にしており，人間のフィードバックを利用した強化学習 (Reinforcement Learning from Human Feedback; RLHF) によって学習されている．RLHFの導入により，ChatGPTは人間の入力に対して高品質な回答を生成する能力を持つようになった．

本節では，ChatGPTによってレビューに対する宿泊施設の返信を生成させ，その内容を予備的に検証する．ここでの検証では，GPT-3.5ベースのChatGPTに対し，以下のプロンプトを与え，返信を生成する．

> 以下は宿泊施設に対するレビューです。これに対して，宿泊施設として返信をしてください。

実際のレビューと，それに対してChatGPTが生成した返信を表4.14に示す．

ChatGPTが生成した返信を見ると，レビューで言及された不満についてある程度は返信できており，またそれに対する解決策も提示していることが分かる．一方で，文の長さを長くすることによって多くの属性に対して言及する傾向が見られ，冗長性に関しては課題が残る．

表 4.14: ChatGPT により生成された返信

レビュー	ChatGPT により生成された返信の例
先日お世話になりました。フロントの方は親切身があり良い印象でした。ただ自分はよくホテルに泊り朝食を必ず食べるのですが、このホテルの朝食はあの値段で内容がだいぶ貧しく思えました。また食事場所も殺風景な宴会場。一日のスタートになる朝食なんですけどとてもガッカリしました。値段に合う分の内容、フインキじゃなくて残念。まだ近くのビジネスホテルの朝食が安くておいしいですよ。次回お世話になるときは改善されてることを期待してます。	ご投稿いただきありがとうございます。お客様には当ホテルをご利用いただき、誠にありがとうございます。また、フロントスタッフの対応に関して好意的なお言葉をいただき、ありがとうございます。一方で、朝食に関するご意見については、大変残念でございます。お客様がご期待に添えない朝食の内容や食事場所の雰囲気についてのご意見、誠に申し訳ございませんでした。お客様のご感想を真摯に受け止め、今後の改善点として十分に検討させていただきます。おっしゃる通り、朝食は一日のスタートを左右する大切なひとときであり、ご満足いただけなかったことを重く受け止めております。今後、より質の高い朝食を提供できるよう努力し、お客様にご満足いただけるサービスをご提供できるよう心がけてまいります。次回のご滞在においては、改善されたサービスでお迎えできるよう、スタッフ一同努力いたします。またお近くのビジネスホテルの朝食に関しても参考にさせていただき、更なる向上を目指してまいります。貴重なご意見をお寄せいただきありがとうございます。お客様のご期待に添えるよう、今後も一層の努力を重ねてまいります。何かご要望やご質問がございましたら、どうぞお気軽にお知らせください。
毎回松戸に来た時は必ず利用させてもらってますが、今回は初めて嫌な気分になりましたそれはテレビを夜中や早朝でも つけていて音量が聞こえるのが複数あったため あまり寝られなかったです 定期的に巡回をしてもらって音が漏れてる部屋に直接注文等をしていただきたいと思いました。	お客様、いつも当ホテルをご利用いただきまして誠にありがとうございます。また、今回のご滞在での不快な経験について、誠に申し訳ございませんでした。おっしゃる通り、他のお客様の行動が快適な睡眠に影響を及ぼすことは避けたいと考えております。特に深夜や早朝においては、周囲のお客様に配慮が求められます。ご指摘いただいたように、テレビの音量については巡回での確認を強化し、漏れている部屋に対して直接的な対応を行えるよう努めてまいります。お客様が安心してご滞在いただけるよう、今後はお客様の声により敏感に対応できるようなシステムの確立に努めてまいります。お客様のご指摘を真摯に受け止め、今後のサービス向上に取り組んでまいります。何かご要望やご質問がございましたら、どうぞお気軽にお知らせください。またのご利用を心よりお待ちしております。

第5章　おわりに

5.1　研究のまとめ

本研究では，宿泊施設のレビューに対して，多様な表現で構成され，かつレビューで言及されている全ての不満に言及する返信を生成することを目的とし，これを実現する手法を提案した．提案手法の手続きは以下の通りである．まず，句点を文の境界としてレビューを分割した．次に，苦情判定モデルにより，レビュー文ごとに苦情か否かを判定し，苦情でない文を除外した．そして，苦情と判定されたレビュー文を入力として，返信生成モデルによって返信を生成した．最後に，レビュー文ごとに生成された返信を統合し，最終的な返信の出力とした．

苦情判定モデルは，BERT をファインチューニングして学習した．楽天トラベルデータセットから，1 つの文から構成される (句点が 1 つしかない) レビューで，苦情ラベルが付与されたレビューとそうでないレビューを同数選別し，苦情判定モデルの訓練データとした．

返信生成モデルは，BART をファインチューニングすることで学習した．楽天トラベルデータセットにおけるレビューとそれに対する返信の組を訓練データとした．さらに，訓練データの質を向上させるため，2 種類の手法でフィルタリングを行った．1 つ目は，ユーザがレビューで述べている宿泊施設の属性に対する返信を生成するために，属性に言及していない返信を訓練データから除外する手法であった．具体的には，属性語がレビューと返信の両方に存在するデータのみを残し，それ以外のデータを削除した．属性語は訓練データからあらかじめ抽出した．全ての単語について，全てのレビューにおける TF-IDF を算出し，その TF-IDF 値の上位 1000 件の平均を単語のスコアとし，そのスコアの上位 500 件の単語を属性語とした．2 つ目は，紋切り型の返信が生成されるのを抑制するために，訓練データにおける返信から定型文を除外する手法であった．宿泊施設の返信を文に分割し，それぞれの文に対して定型度スコアを算出し，その上位 30%の文を定型文として削除した．この処理の後の訓練データは，レビューと，それに対する元の返信から定型文を除いたテキストの組とした．定型度スコアは，レビュー文における単語 tri-gram の，訓練データ全体における出現頻度の平均とした．

返信の統合では，重複する表現を除外するため，2 つの返信文間の距離を正規化された編集距離で測り，それが 0.1 以下の場合，元のレビューにおける出現順序が後である返信文を残し，もう一方の返信文を除外した．ただし，属性語を含む返信は常に除外しないとした．残された返信文を元のレビューと同じ順序で連結し，

最終的な返信とした．

提案手法の評価実験では，苦情判定モデルと返信生成モデルを個別に評価した．苦情判定モデルは，構築したデータセットを訓練データとテストデータに分け，テストデータにおける苦情判定の正解率やF値などを算出した．その結果，BERTをファインチューニングする際のepoch数が1の時に正解率とF値が最も高く，それぞれ0.8877と0.8901となり，苦情判定モデルの性能が十分に高いことを確認した．

返信生成モデルについては，自動評価と人手評価を行った．自動評価では，訓練データに対するフィルタリングの有無や文毎に返信を生成するか否かなどで分けた6つの手法を比較し，生成した返信のBLUE-4とDISTINCT-4を算出した．BLEU-4はフィルタリングを行わないベースラインが一番高くなった．これはフィルタリング処理をしないために定型文が生成されることが多いが，評価データにおける正解の返信にも定型文が多く，両者で単語n-gramが一致することが多いためと考えられた．DISTINCT-4はフィルタリング処理を行う手法の方が高く，定型文のフィルタリングにより，ありきたりな文の生成が抑制された．人手評価では，返信の流暢性，非冗長性，総合評価，属性言及率を評価した．その結果，ベースラインよりも属性に言及しない返信のフィルタリングを行う提案手法のほうが属性言及率が高くなった一方で，流暢性と非冗長性のスコアはベースラインと比較して低くなった．属性言及率と流暢性・非冗長性はトレードオフの関係にあることがわかった．定型文のフィルタリングに着目すると，非冗長性が改善され，流暢性や総合評価も高くなり，紋切り型の表現の生成が抑制されていることが確認できた．また，属性が複数あるレビューに関しては，文毎に返信を生成する手法はそうでない手法に比べて属性言及率と総合評価が高く，その有効性が確認できた．

提案手法の特徴を明らかにするために，ベースラインと提案手法によって生成された返信を比較し，属性に言及しない返信のフィルタリング，定型文のフィルタリング，文毎の返信を生成して統合する機構の導入によって，望ましい返信が得られている事例を紹介した．また，あるレビューに対し提案手法によって返信を生成する過程を説明し，レビュー文の分割，苦情判定，返信の統合処理のそれぞれについて，これらが有効に働く例や不十分である例を紹介した．

5.2 今後の課題

評価実験では提案手法の有効性が確認できたが，全指標において実際の宿泊施設の返信(GOLD)の評価値が最も高く，これを超えることはできなかった．そのため，提案手法には改善の余地がまだ多く残されていると言える．

人手評価においては，文を分割する手法はそうでない手法に比べ属性言及率が上がる一方，非冗長性や流暢性が下がり，全体として総合評価が低下する傾向にあった．そのため，属性言及率は維持しつつ，非冗長性や流暢性を上げるような返信の統合処理の手法を考案する必要がある．本研究における統合処理は，文間の編集距離を計算することで類似した文を除外しているだけなため，結合した文

同士の繋がりが不自然であることも多い．そのため，結合した文を接続詞でつなぐようにしたりするなど，より自然な返信を生成するように文を結合する方法を探求する必要がある．

4.4 節でも説明した通り，ユーザは 1 つの属性に関する不満を 1 つの文だけではなく複数の文によって表明することもある．そのため，レビューを文に分割する代わりに，1 つの属性に言及しているパッセージに分割し，個々のパッセージに対して返信を生成して，これらを統合することが対策として考えられる．ここでのパッセージとは，文の一部であるときもあれば，複数の文をまとめたテキストになることもある．1 つの文に複数の属性が含まれていればそれを別々のパッセージに分割したり，1 つの属性が複数の文にまたがって言及されていればそれらをまとめて 1 つのパッセージとしたりするなどの処理を検討する．

返信生成モデルについても，ベースラインに比べ不満の属性に言及できるようになり，定型文の出現も抑制できるようになったものの，依然として「〜につきましては」や「〜の件では」といったパターンのような表現が頻出する傾向が見られた．その対策として，モデルが生成した返信に対して，特定のフレーズや表現を検出し，置換または除去するフィルターを導入するという方法が挙げられる．これにより，生成された返信から冗長な表現を取り除き．自然さと多様性を向上させることができると考えられる．

謝辞

本研究を進めるにあたり，献身的な支援と専門的な指導を賜りました主指導教員である白井清昭准教授に心からの感謝の意を表します．また，副指導教員である井之上直也准教授，白井研究室のメンバーにも感謝申し上げます．

参考文献

[1] BART 日本語 Pretrained モデル – LANGUAGE MEDIA PROCESSING LAB, (2023-12 閲覧). https://nlp.ist.i.kyoto-u.ac.jp/?日本語 Pretrained モデル.

[2] BERT base Japanese (IPA dictionary, whole word masking enabled) – Hugging Face, (2023-12 閲覧). https://huggingface.co/cl-tohoku/bert-base-japanese-whole-word-masking/.

[3] AdamW (PyTorch), Transformers – Hugging Face, (2024-1 閲覧). https://huggingface.co/docs/transformers/main_classes/optimizer_schedules#transformers.AdamW.

[4] Introducing ChatGPT, (2024-1 閲覧). https://openai.com/blog/chatgpt.

[5] 【無料 dl 資料】宿泊予約経路の構成比率で倍になったのは？集客対策ベスト 5 発表「ネット予約・検索対策等の状況調査」 - 旅館ビジネスの「今」をキャッチする／リゾ lab【リゾラボ】, （2023-12 閲覧）. https://www.resort-lab.com/1106/.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4171–4186, 2019.

[7] Joseph L. Fleiss and Jacob Cohen. The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and Psychological Measurement*, Vol. 33, No. 3, pp. 613–619, 1973.

[8] Cuiyun Gao, Jichuan Zeng, Xin Xia, David Lo, Michael R. Lyu, and Irwin King. Automating app review response generation. In *2019 34th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pp. 163–175, 2019.

[9] Tannon Kew, Michael Amsler, and Sarah Ebling. Benchmarking automated review response generation for the hospitality domain. In *Proceedings of Workshop on Natural Language Processing in E-Commerce*, pp. 43–52, 2020.

[10] Tannon Kew and Martin Volk. Improving specificity in review response generation with data-driven data filtering. In *Proceedings of the Fifth Workshop on e-Commerce and NLP (ECNLP 5)*, pp. 121–133, 2019.

[11] Vladimir I Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. *In Soviet physics doklady*, Vol. 10, pp. 707–710, 1966.

[12] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7871–7880, 2020.

[13] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 110–119, 2016.

[14] Rakuten Institute of Technology. 楽天データ公開. `https://rit.rakuten.com/data_release_ja/`. (2024年1月閲覧).

[15] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311–318, 2002.

[16] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[17] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, Vol. 1, No. 8, p. 9, 2019.

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, p. 6000–6010, 2017.

[19] Lujun Zhao, Kaisong Song, Changlong Sun, Qi Zhang, Xuanjing Huang, and Xiaozhong Liu. Review response generation in e-commerce platforms with external product information. In *The World Wide Web Conference*, p. 2425–2435, 2019.

[20] 伊草久峻, 鳥海不二夫. レビュー特性を用いたレビュー返信の自動生成. 人工知能学会全国大会 (第 35 回) 論文集, pp. 2F3–GS–10g–01, 2021.