

宿泊予約サイトにおけるレビュー返信の自動生成

Automating Review Response Generation on Hotel Reservation Sites

伊草久峻 鳥海不二夫
Hisatoshi Igusa Fujio Toriumi

東京大学 大学院工学系研究科システム創成学専攻
Department of Systems Innovation, School of Engineering, The University of Tokyo

要旨：宿泊予約サイトにおいて、ユーザのレビューや評価、またそれに対するホテル側からの返信は極めて重要な情報であり、ユーザはそれらを参考にしてホテルの予約を行う。そのためホテル側はユーザのレビューに適切に返信を行うことが顧客獲得のために重要となる。しかし全てのレビューに対応することは従業員にとって大きな負担であり、返信へのサポートが求められている。本研究では、Decoderでの学習の仕方を変えた2種類のRNN Encoder-Decoderモデルを用いて、ユーザのレビューからホテル側の返信を自動生成するモデルを作成した。さらに、より滑らかな文章にするために単語の生起確率により補正する機構をモデルに組み込み、モデルの改良を行った。

キーワード：レビュー、返信生成、RNN Encoder-Decoder モデル

Abstract: On the hotel reservation site, user reviews, evaluations and the hotel's response to them are extremely important information, and users make a hotel reservation with reference to these. Therefore, it is important for the hotel to respond appropriately to user reviews in order to acquire customers. However, responding to all reviews can be a burden on hotel employees, so support is required. In this research, we create a model that automatically generates hotel's responses from user reviews using two types of RNN Encoder-Decoder models that changed the way of learning with Decoder. Furthermore, in order to make the sentence smoother, we improved the model by incorporating a mechanism that corrects by the probability of occurrence of words into the model.

Keyword: review, response generation, RNN Encoder-Decoder model

1. はじめに

近年、インターネットの普及により、宿泊施設の予約はオンラインの宿泊施設予約サイトを通じて行われることが増えてきている。以前は宿泊施設へ直接電話する、旅行会社に向く等の手段が一般的であったが、宿泊予約サイトでは、目的に合致した宿泊施設を容易に検索できるだけでなく、複数の宿泊施設を比較検討することが容易であるなど、利用者にとって利点が多い。さらに、実際に宿泊したユーザからのレビュー（口コミ）機能があるサイトも多く、これらのレビュー文はサイト利用者にとって宿泊施設を決めるうえで、有用な情報となっている。

また、宿泊施設側にとってもレビュー文は宿泊者の声を直接聞く重要な機会である。特に、宿泊者の評価が低い場合は口コミ内容からその理由を探り、適切に対処することで改善を目指すことも可能である。逆に、低い評価のレビューを放置しているとユーザ側から不信感を持たれ、見込み顧客を逃すことになる可能性もある。そのため、レビューに対して適切に対応することは宿泊者数を増やすうえでは重要であると考えられる。

一方で、日々の業務に追われる宿泊施設の従業員にとって返信を含むレビューへの対応は負荷となるため、レビュー対応がおろそかになるケースも多い。そのためレビュー対応の有用性を確認し、さらに実際にレビューへの返信のサポートを行うことが求められている。

本研究の目的は、宿泊予約サイトに書き込まれたレビューに対して適切な返信例を自動で作成するシステムを実現させることである。本論では、RNN Encoder-Decoder モデルを用い

て、学習の方法を変えながら最適な返信文生成モデルを模索する。

2. 関連研究

レビューや口コミの重要性が認識されて以来、レビュー分析において多数の研究が行われている。特に、レビュースコアの不正発見は様々な研究が行われている。Feng ら [1] は、不正スコアによるレビュースコアの分布の歪みを確認し、レビューの代表的な分布の定量的評価を行った。Narayan ら [2] は、機械学習を用いてレビューのスパム検出を行い、さまざまな分類モデルについてその評価を行った。Junting ら [3] は、製品がスパムのターゲットとなる可能性を定量化する新たな尺度の提案、スパムのクラスタ化を行い、スパムグループとその標的製品の発見手法を提案した。

また、レビュー文の要約を行う研究も多数行われている。小池ら [4] は、重要文抽出によるレビューの要約における問題点を指摘し、重要文抽出の前処理である重要文候補という概念を定義し、その抽出手法を提案した。立石ら [5] は、Web 文書から意見を抽出し、それらをレーダーチャートの形式で要約・視覚化する意見抽出分類システムを提案し、人手で作成したレーダーチャートと比較的一致することを確認し、分析の起点となることを示した。

他にも、成川 [6] は、苦情データを対象にコーディング機能を利用したテキストマイニングを行うことで各宿泊施設の苦情の原因を明らかにするとともに、和風・洋風などといった宿泊施設自体が持つ属性の違いや宿泊施設が存在する地域の違いをもとに苦情の特徴を分析した。また内田ら [7] は、レビュー文のオノマトペを用いて商品のカテゴリを類型化している。

一方、レビューへの返信に関する研究は多くはない。例え

連絡先: 伊草久峻, 東京大学

大学院工学系研究科システム創成学専攻,
東京都文京区本郷 7-3-1, igusa@torilab.net

ば, Sparks ら [8] は, 旅行口コミサイトとして世界最大の閲覧数を持つ, TripAdvisor^{*1} における口コミへの返信がどのように行われているのかを分析し, Acknowledge, Account, Action のそれぞれが含まれる割合を示すトリプル A モデルを提案し, ランキング上位下位ホテルでの違いを明らかにした. また Xie ら [9] も TripAdvisor における口コミへの返信を分析し, ホテルのランクと口コミ返信および RevPAR(客室平均単価 x 客室稼働率) を用いてホテルのランクに応じた最適な返信パターンを発見した. これらはいずれも TripAdvisor のデータを用いた最適な返信文を明らかにする研究である.

また対話の生成については, Microsoft XiaoIce チャットボット [10] など, 社会的会話を促進するために, 自然言語処理の分野で広く研究されてきた. このような研究は, 一般に RNN Encoder-Decoder モデル [11] に基づいており, コンテキストとそれに対応する応答をそれぞれソースシーケンスとターゲットシーケンスと見なして学習を行う. RNN Encoder-Decoder モデルは特に自動翻訳に用いられている学習アプローチである. また, 自然言語クエリを指定してアプリケーションのシーケンスを生成したり [12], 自然言語をマシンで解釈可能なシーケンス (データベースクエリなど) に解析したり [13], コミットメッセージを生成したりする [14] など, 多くのソフトウェアエンジニアリングタスクに適用されている.

3. 対象データの概要

本研究は, 国立情報学研究所の情報学研究データリポジトリにある, 楽天株式会社 の楽天トラベルデータよりユーザレビューとそれに対するホテル側からの返信文を対象とした.

またデータには, 施設番号や投稿番号, 投稿日時, ユーザ評価などの情報も含まれている.

本研究ではユーザレビューとホテル返信文が揃っているデータについて分析, 研究を行うため, それらが揃っている 5,348,057 件のデータを対象に, ユーザレビューをソースシーケンス, ホテル側返信文をターゲットシーケンスとみなして分析を行った. 対象となったデータの期間は 2004 年から 2019 年までとなっている.

4. RNN Encoder-Decoder モデルを用いた返信文生成

4.1 モデル概要

RNN Encoder-Decoder モデルを用いてユーザレビューからホテル側返信文の生成を行う. モデルの概要は図 2 のようになっている. RNN Encoder-Decoder モデルは, ソースシーケンス $x = (x_1, x_2, \dots, x_{T_x})$ が与えられたときに, ターゲットシーケンス $y = (y_1, y_2, \dots, y_{T_y})$ の生成を目的とする. ここで T_x と T_y は, それぞれソースとターゲットのシーケンス長である. まず Encoder においてソースシーケンス x を隠れベクトル h_1, h_2, \dots, h_{T_x} のセットに変換する. また, コンテキスト表現 c は Recurrent Neural Network (RNN) を用いて生成される. このようにして Encoder RNN は, 次式のようにソースシーケンスの最初のトークンから最後のトークンまでを順に読み取る.

$$\begin{aligned} h_t &= f(h_{t-1}; w_t) \\ c &= h_{T_x} \end{aligned}$$

ここで w_t はソーストークン x_t の単語埋め込み表現であり, 本研究では日本語版 Wikipedia で学習済みの word2vec[17] を用いて変換した 200 次元のベクトル表現を使用した. f は前の隠れ状態 h_{t-1} を考慮しつつ, 単語埋め込み表現 w_t を隠れ状態 h_t にマッピングする非線形関数である.

次に Decoder においては, Encoder と同様に RNN として実装し, 次式において隠れ状態 h'_t と前の単語 y_{t-1} に基づいて, タイムスタンプ t ごとに 1 単語 y_t を生成する.

$$P_r(y_t | y_{t-1}, \dots, y_1, c) = g(h'_t, y_{t-1}, c) \quad (1)$$

ここで g は非線形マッピング関数であり, Encoder によって返されるコンテキストベクトル c は最初の隠れ状態として $h'_1 = c$ と設定する. Decoder では, シーケンスの終わりの単語を生成すると停止する.

2 つの RNN Encoder-Decoder モデルは, 次式のように条件付き対数尤度を最大化するように共同でトレーニングを行う.

$$L(\theta) = \max_{\theta} \frac{1}{N} \sum_{i=1}^N \log p_{\theta}(y_i | x_i) \quad (2)$$

ここで θ はモデルのパラメータセット (ニューラルネットワークの重みなど) であり, 各 (x_i, y_i) はトレーニングセットのソースシーケンスとターゲットシーケンスのペアとなっている. $p_{\theta}(y_i | x_i)$ は, ソースシーケンス x_i が与えられた場合にモデルのパラメータに従って i 番目のターゲットシーケンス y_i を生成する可能性を表す. 最適化アルゴリズムを使用して損失関数を最適化することによって, パラメータ θ の最適値の推定を行う.

本研究では Decoder での学習の方法を 2 通りに変えて行い比較した. 各モデルについては 4.2 節と 4.3 節で述べる. さらに, 4.4 節で述べるように, 単語の生起確率を組み込んだモデルを実装した. すべてのモデルについて共通する設定として, RNN に GRU を使用し, 単語の埋め込み次元を 200 次元, Encoder と Decoder の隠れベクトルのサイズを 200 次元, バッチサイズを 32 とした. これらのモデルについて生成文を比較, 検討した.

4.2 単語のベクトルを学習する手法

1 つめの手法として, Decoder で単語を生成するにあたり word2vec での単語埋め込み表現を学習する手法を考える. word2vec の埋め込み表現は, Encoder と同様に日本語版 Wikipedia で学習済みの word2vec[17] を用いた. ターゲットシーケンスの各単語について word2vec のエンベディングベクトルを学習し生成するように, 次式の loss 関数を最小化させる.

$$\text{loss} = \sum_{i=1}^N 1 - \text{cossim}(w_i, w_{p_i}) \quad (3)$$

ここで w_{p_i} はモデルによって生成された単語の埋め込み表現であり, $\text{cossim}(w_i, w_{p_i})$ は i 番目のターゲットシーケンスの単語の埋め込み表現と生成した単語の埋め込み表現の cos 類似度を表す.

単語の生成としては出力した単語ベクトルに最も近い単語ベクトルの単語を選ぶ.

4.3 語彙中の単語から学習する手法

2 つめの手法として, あらかじめ用意した語彙の中からの単語かを Decoder で学習する手法を考える. 語彙はターゲット

*1 <https://www.tripadvisor.com/>

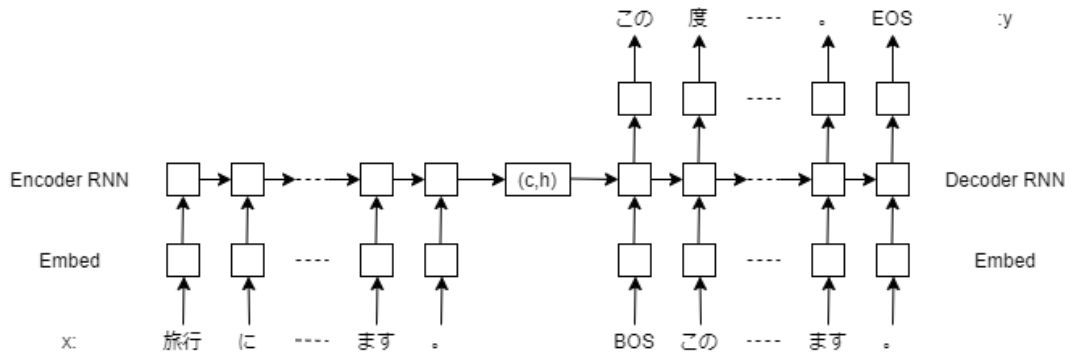


図 1: RNN Encoder-Decoder モデル概要

トシーケンスの学習データから作製し、語彙数はデータの登場頻度順に上位 4 万単語を用いた。ターゲットシーケンスの各単語について語彙中の単語のうちのどれであるかを学習し生成するようにし、次式の loss 関数を最小化させる。

$$loss = - \sum_{i=1}^N p(y_i) \log p_{\theta}(y_i|x_i) \quad (4)$$

ここで $p(y_i)$ はターゲットシーケンスの各単語の真の値を表し、 $p_{\theta}(y_i|x_i)$ はモデルにより予測した単語の確率分布を表す。

単語の生成としては語彙中の確率分布のうち最も確率の高い単語を選ぶ。

4.4 単語の生起確率を組み込んだ手法

日本語の文章を生成するにあたり、日本語としてもっともらしい単語の並びである必要がある。さらに、前の単語から次の単語を予測するうえで単語の並びの確からしさは一つの情報になると考えられる。

そこで単語の並びの情報をモデルに加えてモデルを改良するために、語彙中の一つ前の単語から次の単語の生起確率をモデルに組み込む。単語の生起確率はターゲットシーケンスの学習データから計算し、図で示すように、4.3 のモデルの Decoder での各単語の隠れベクトルに、一つ前の単語からの生起確率のベクトルを結合し、単層のニューラルネットワークにより学習を行う。最終的な学習の対象は 4.3 のモデルと同様であり、loss 関数も同様に式 (4) により学習を行う。

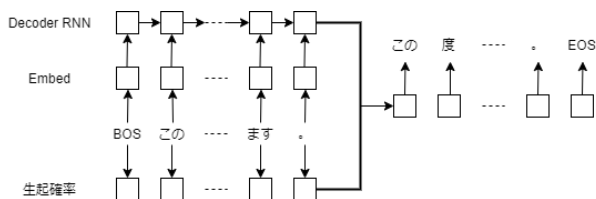


図 2: 単語の生起確率を組み込んだモデルの Decoder 概要

4.5 各手法による生成例

4.2 節, 4.3 節, 4.4 節の 3 つのモデル全てについて、ユーザーレビューとホテル側返信文のデータセットのうち 7 割を学習データ、3 割をテストデータとして学習を行った。

各モデルによって生成したレビューへの返信文の例を表 1 に示す。モデル 1, 2, 3 はそれぞれ、4.2 節, 4.3 節, 4.4 節のモデルを表す。

4.6 考察

まずモデル 1 とモデル 2 の生成例について比較を行う。モデル 1 の生成例においては、表 1 の例 1 で「ホテル」という単語が続いてしまったり、例 2 で本来であれば考えにくい場所で「お客様」という単語を生成してしまっていたりすることが確認される。モデル 2 においてはモデル 1 に比べ、このように同じ単語を頻繁に生成することが少なく、特に表 1 の例 1 のように定型的で短めの返信文生成については比較的正確に生成できていることから、モデル 2 の方がレビューへの返信の学習ができていていると考えられる。これは、各単語自体がどれであるかを直接学習するのではなく単語のベクトルを推定するというモデル 1 の仕様において、単語のベクトルをピンポイントで当てるのは非常に困難であり、loss を低く抑えるためにあらゆる単語との cos 類似度が小さくなるような平均的なベクトルを出力するような学習が行われたことが一因としてあるのではないかと考えられる。

次にモデル 2 とモデル 3 の生成例について比較する。表 1 の例 1 の生成例においてはどちらも同程度の正確さであると考えられるが、ターゲットであるホテル側返信文と比較したときに、文の最後にある「お待ちしております」という文言の表現に違いが見られる。モデル 2 では「お待ちしております」となっておりほぼ同じ表現であるが、モデル 3 では「お待ち申し上げます」となっており単語数も違う表現となっている。これはモデル 2 においては単純に学習が上手く行われたからであると考えられる。一方モデル 3 においてはターゲットの表現から変わっているものの、日本語として妥当であり同じ意味の表現を生成できていることから、単語の生起確率の情報から日本語としてふさわしい単語の並びでの生成や補正が行えていることがわかる。ここから、単語の生起確率を組み込むことにより、より柔軟に日本語らしい文章を生成でき、単語の生起確率を考慮することに有用性があると考えられる。また、表 1 の例 2 の生成例においても生成文の後半部分についてはモデル 3 の方が比較的日本語として自然な文章になっていると考えられるが、その一方特にターゲットシーケンスの第 2 文にあたる部分の生成文についてモデル 2, モデル 3 ともに正解の文章とかなり異なっており、また日本語としても成立していない文章になってしまっている。これは、ユーザーレビューの「合格」という文言に対して直接的に応答するというやりとりをうまくモデルで再現できていないためであると考えられる。このように、レビューごとの要点となるようなポイントやキーワードとなる単語は返信文を生成するうえで大事な情報となると考えられ、これらの情報を明示的にモデルに組み込んで考慮

表 1: ユーザレビュー, 返信文と各モデルの生成例

例 1
<p>ユーザレビュー: 最初はホテルの場所が解り辛く、一度市駅まで戻ってから帰りました。翌日明るくなってみれば実は駅からはかなり至近だった事がわかります。道路が直角に交差していないのと川が直角に蛇行(とはいえないが)しているので距離感がつかめなかったのでしょうか? 第1との微妙な位置も一因でしょうか? 禁煙ルームありでリーズナブルですのでまた利用したいと思います。館内のレストランもメニューが豊富そうなので次回は利用したいですね。</p> <p>ホテル側返信文: この度は、(ホテル名)をご利用いただきまして誠に有難うございました。これからもご満足頂けますよう努力して参ります。又のご利用をお待ちしています。フロント 石倉</p> <p>モデル 1: この度は、ごホテルホテル 1 (ホテル名) をご利用頂きまして誠にありがとうございます。またもお客様利用いただけますよう、して参ります。またのご利用を心申し上げます。</p> <p>モデル 2: この度は、当 (ホテル名) をご利用頂きまして誠にありがとうございます。またも、満足頂けたよう、して参ります。またのご利用をお待ちしております。</p> <p>モデル 3: この度は、(ホテル名) をご利用いただきまして誠にありがとうございます。またもお客様満足いただけるますよう、して参ります。またのご利用をお待ち申し上げます。</p>
例 2
<p>ユーザレビュー: 今回初めて宿泊しました。料金からあまり期待していませんでしたが、この料金で部屋の設備、朝食無料、備品の充実、から考えると合格です。</p> <p>ホテル側返信文: この度は、当ホテルをご利用くださいませ誠にありがとうございます。合格点を頂きまして、恐縮でございます。まだまだ至らぬ点もございますが、お客様のご意見を参考に、更にご満足頂けます様、スタッフ一同努力してまいります。金沢にお越しの際は、ぜひ当ホテルをご利用下さいませ。スタッフ一同、お待ち申し上げます。</p> <p>モデル 1: この度は、ごホテルをご利用頂きまして誠にありがとうございます。またのございました、お客様でございます。今後のないことがあります。お客様にご意見をいただきにさ今後安心満足いただけますよう、努力一同同努力して参ります。またにお越しの際は、是非お客様ホテルをご利用下さいませ。お客様一、心申し上げます。</p> <p>モデル 2: 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。またしておました、重ねでございます。今後至らぬ点がございませ、お客様にご意見を参考にさスタッフ努力満足頂けたよう、スタッフ一同努力して参ります。またにお越しの際は是非是非ごホテルをご利用下さいませ。また一、心申し上げます。</p> <p>モデル 3: この度は、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。また室がおました、誠にございます。また至らぬ点がございませ、ごにご意見を参考にさお客様一層満足いただけるますよう、スタッフ一同同サービスして参ります。またにお越しの際は是非是非ごホテルをご利用下さいませ。また一、心申し上げます。</p>

できるようにすることでモデルの改良が図れると考えられる。そのために、レビューの要点や重要であると考えられる文言を抽出したキーワードや、宿泊予約サイトにおいてレビューとともに投稿される、「立地」や「食事」といった項目ごとの評価値などを返信文の生成モデルに組み込むことが考えられる。

また、全体に「ます」と「まし」などの活用による間違いが散見される。これらは活用の違いのみであり、文章の意味的部分ではなく文章としての滑らかさにかかわる部分であると考えられるため、これらを2つの段階に分けて考え、生成段階ではまず活用を考慮しない原形に直してトレーニングし、2段階目で原形のみで生成された文章を滑らかな文章にするように活用させるという手法が考えられる。これにより生成段階においては語彙数も減らすこともでき、潜在的な意味が同じ単語で学習が困難になることを防ぐことが出来るのではないかと期待される。

さらに、本論では各モデルの評価として返信の生成例を実際に目で見て確認したが、確認できている数も多くなく指標として不確かであるため、まず機械的に多くのテストデータを確認しモデルの優劣を比較する必要があると考えられる。そのために文章要約の自動評価手法としても知られる BLUE などの評価手法を適用することが考えられる。

5. 結論

本研究では、宿泊予約サイト「楽天トラベル」のユーザレビュー、ホテル側返信文を用いて、レビューの文章から返信文の自動生成をするモデルの構築を行った。

Decoder での学習の手法として、単語の埋め込み表現を学習するモデルと語彙中の単語から推定するモデルを構築し比較したが、後者の方が適切に学習できている様子が確認できた。これは前者に比べ後者のモデルの方が単語をピンポイントで学習できる仕組みであるからだと考えられる。さらに、語彙中の単語から予測するモデルに単語の生起確率を組み込んだモデルを構築し、単語の並びの情報が文章生成において有用性があることを確認した。しかしどのモデルも実用的に使えるような返信文生成には及ばず、改良の余地が多く残されている。

今後の課題としては、返信文の生成をよりの確で滑らかな文章にするために、レビュー中のキーワードやユーザの評価などのレビュー固有の特性をモデルに組み込む、単語の活用を文章生成に分けるなどの改良を行うことが考えられる。さらにより機械的により良いモデルを判定するために、BLUE などの文章の自動評価手法の適用が考えられる。

謝辞

本研究は、楽天株式会社 (2020): 楽天トラベルデータ. 国立情報学研究所情報学研究データリポジトリ. (データセット). <https://doi.org/10.32130/idr.2.2> を用いて行ったものである。データを提供していただいた楽天株式会社と国立情報学研究所に感謝いたします。

参考文献

- [1] S Feng, L Xing, A Gogar, Y Choi: Distributional footprints of deceptive product reviews. Sixth International AAAI Conference on Weblogs and Social Media(2012)
- [2] Narayan, Rohit, Jitendra Kumar Rout, Sanjay Kumar Jena: Review spam detection using opinion mining, Progress in Intelligent Computing Techniques: Theory,

-
- Practice, and Applications. Springer, Singapore, 273-279, (2018)
- [3] Ye, Junting, Leman Akoglu: Discovering opinion spammer groups by network footprints, Joint European Conference on Machine Learning and Knowledge Discovery in Databases. (2015)
- [4] 小池惇爾, 松吉俊, 福本文代: 評価視点別レビュー要約のための重要文候補抽出, 言語処理学会第 18 回年次大会論文集 pp.1188-1191 (2012)
- [5] 立石健二, 福島俊一, 小林のぞみ, 高橋哲朗, 藤田篤, 乾健太郎, 松本裕治: Web 文書集合からの意見情報抽出と着眼点に基づく要約生成. 情報処理学会研究報告情報学基礎 (FI), pp.1-8.(2004)
- [6] 成川健太郎. "楽天トラベルにおける苦情データのテキストマイニングに関する研究." KGPS review: Kwansai Gakuin policy studies review 26 (2019): 15-22.
- [7] 内田ゆず, 荒木健治: クラスタ分析を用いた商品レビューに含まれるオノマトベに基づく商品カテゴリーの類型化, 人工知能学会論文誌 Vol.30 No.1 pp.246-256 (2015)
- [8] Sparks, Beverley A., Graham L. Bradley: A "Triple A" typology of responding to negative consumer-generated online reviews, Journal of Hospitality & Tourism Research Vol.41 No.6 719-745(2017)
- [9] Xie, K., Kwok, L., Wang, W. : Monetizing managerial responses on TripAdvisor: Performance implications across hotel classes. Cornell Hospitality Quarterly, Vol.58 No.3, pp.240-252.(2017)
- [10] L. Zhou, J. Gao, D. Li, and H. Shum: "The design and implementation of xiaoice, an empathetic social chatbot," CoRR, vol. abs/1812.08989,2018.
- [11] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio: "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, 2014, pp. 1724-1734.
- [12] X. Gu, H. Zhang, D. Zhang, and S. Kim: "Deep API learning," in Proceedings of the 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering, FSE 2016, Seattle, WA, USA, November 13-18, 2016, 2016, pp. 631-642.
- [13] L. Dong and M. Lapata: "Language to logical form with neural attention," in Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers, 2016.
- [14] S. Jiang and C. McMillan: "Towards automatic generation of short summaries of commits," in Proceedings of the 25th International Conference on Program Comprehension, ICPC 2017, Buenos Aires, Argentina, May 22-23, 2017, 2017, pp. 320-323.
- [15] Kudo, Taku: Mecab: Yet another part-of-speech and morphological analyzer: <http://mecab.sourceforge.net/>(2005)
- [16] 鈴木正敏, 松田耕史, 関根聡, 岡崎直観, 乾健太郎: 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)
- [17] 鈴木正敏, 松田耕史, 関根聡, 岡崎直観, 乾健太郎: 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)