

格フレームを用いた質問生成によって対話の掘り下げを行う対話システム

古川 智雅¹ 吉野 幸一郎^{1,2} 須藤 克仁¹ 中村 哲¹

¹奈良先端科学技術大学院大学情報科学研究科

²科学技術振興機構 さきがけ

{furukawa.tomomasa.fo1, koichiro, sudoh, s-nakamura}@is.naist.jp

1 はじめに

近年、スマートフォンやスマートスピーカーといったデバイスや家電製品、自動車等において、人間と対話を行う、対話システムが注目を集めている。この中で、対話を通じてユーザが欲しい情報を提示する情報検索や、ユーザとシステムが雑談対話を行う技術が開発されてきている。しかし一方で、現在の対話システムでは、一問一答を越えてより内容に踏み込んだ対話を行うことが難しい。これは、対話の流れを踏まえ、どの部分に焦点を当てて対話を掘り下げるかということが考慮されていないためと考えられる [1]。

ここで言う対話の掘り下げとは、話題や焦点を絞り対話を継続する行為である。対話の掘り下げにはシステムからの質問が有効であることが知られている [2]。システムが行う質問には特定の目的を達成するためのものと、次のユーザの発話を促すためのものがある。前者は、レストラン検索 [3] や文章理解のための質問生成 [4] といった、特定の達成目標を持った対話システムで、主にユーザから情報を得るために用いられている。これに対し後者は、主にユーザと長く対話を行うことに主眼を置く雑談対話 [5] や傾聴対話 [2] といった、対話を行うことそのものを目的としている対話システムで用いられている。

本研究では、発話を促すための質問と情報を獲得するための質問の、いずれの質問も生成することが可能な対話システムを構築することを指向している。タスクに依存しない質問生成の枠組みを構築することで、情報獲得を行いつつ対話の継続を目的とするような対話システムの構築を目指す。タスクやドメインにかかわらず質問生成を行うため、格フレーム [6] を用いて質問文を生成する対話システムを提案する。また提案システムを評価するためのテストセットを作成し、そのテストセットを用いて提案システムの評価を行った。

2 関連研究

対話システムにおける質問生成機能を達成する手法として、既存研究では主に二つの手法が提案されている。一つはルールによる質問生成で、人手で設計された質問生成のルールに基づいて質問文を生成する。下岡ら [2] は傾聴対話と呼ばれるユーザに発話を促す対話システムにおいて、述語ごとに質問生成を行う必要がある要素をまとめた辞書を人手で作成し、それをもとに質問文を生成した。こうした手法は、文法の誤りや意味の誤りを含んだ質問文を生成しにくいという長所や、目的に合わせた質問文を生成しやすいという長所がある一方、質問生成のルールの設計にコストがかかってしまうという短所がある。

もう一つは統計モデルによる質問生成で、対話コーパスを用いて質問生成モデルを学習させることで、入力文に対して一対一の関係で質問文を生成する。近年では統計的手法として、ニューラルネットワークが盛んに用いられている。Wang ら [5] は疑問詞と質問文の中心となる焦点語を考慮することで、“*Which department are you from?*” などの具体的な質問を生成する手法を提案した。ニューラルネットワークによる手法では、様々なドメインの対話コーパスを用いて質問生成モデルを学習することで、ドメインによらない多様な質問文を生成可能という長所がある一方、文法的な誤りや文としての不自然さが現れやすいという短所や、学習の際に発話と質問がペアとなったデータが大量に必要であるという短所がある。

そこで本研究では、Web から収集した約 5 億文からなる大規模コーパスを用いて構築された格フレーム [6] を用いる。格フレームは、用言ごとに格と要素が格納された、ドメインに依存しない大規模な辞書である。既存の言語資源で、かつ大規模なコーパスから構築されている格フレームを用いることで、人手による複雑なルールの設計や大量の学習データを必要とせずに質問生成を行うことを目的とする。

3 提案手法

本研究が提案する対話システムでは、発話から抽出された用言と、その用言に対して存在する格フレームをもとに質問候補の生成を行った後、質問候補から不要な質問の削除を行う。これは場面や状況に応じた質問生成を行うためである。まず質問文を質問候補として複数生成したのち、いくつかの質問文を削除することで、場面に合わせた適切な質問の生成に結びつけることができる。例えば、「古い昔ながらの家とかですね、なんかもありまして、結構有名な散策コースになってますんでね。」という入力発話例に対して、「どこに有りますか?」、「誰に有りますか?」、「何で有名ですか?」などといった質問候補を生成したのち、「誰に有りますか?」という質問候補を削除する。これは、昔ながらの家の場所に対する質問であれば「どこに有りますか?」、散策コースを有名にしているものに対する質問であれば「何で有名ですか?」といった質問文が考えられる一方、「誰に有りますか?」は質問として不自然と考えられるためである。

3.1 格フレームを用いた質問候補の生成

以下に日本語解析ツールと格フレームを用いた質問候補の生成手順を示す。ここでは、入力発話が「古い昔ながらの家とかですね、なんかもありまして、結構有名な散策コースになってますんでね。」(以下の手順では入力発話例と呼ぶこととする)である場合の、それぞれの手順での出力例も併せて示す。

手順1 格フレームを用い意味カテゴリと格のペアを抽出
まずユーザ発話に含まれる用言に対して、KNP¹を用いた格・構文解析結果により、発話文に含まれない格を抽出する。その後、これらの発話文から検出されなかった格に対して格フレームを参照する。これにより、当該用言に対し存在し得る格要素の候補を得ることができる。これらの格候補に対して形態素解析器JUMAN++²で意味カテゴリを付与することで、意味カテゴリと、格フレームのもととなったコーパス上での格フレーム頻度が得られる。したがって、この段階での入力発話例に対する出力結果は、「ある」という用言に対しては<場所-施設, ニ>、<人, ニ>などのペア、「有名だ」という用言に対しては<人工物-その他, デ>などのペアに、それぞれ格フレーム頻度が与えられたものである。ここで格の抽出の際に用いる格は、ガ格・ニ格・ヲ格・ト格・デ格・

カラ格・ヨリ格・マデ格・ヘ格・ノ格の10個の格に限定する。

手順2 意味カテゴリを疑問詞に置き換えて疑問詞と格のペアを取得

JUMAN++で得られる、名詞の形態素に付与される意味カテゴリ22種³と、事前に作成した疑問詞を一对一で対応付けした辞書を用いてカテゴリを疑問詞に置き換えることで、疑問詞と格のペアを取得する。このとき、手順1で出力された<意味カテゴリ, 格>のペアでは異なっても、手順2で出力された<疑問詞, 格>のペアが同じになった場合、それらの格フレーム頻度を足し合わせて、<疑問詞, 格>のペアの頻度とした。この結果に対し、頻度が多い意味カテゴリほど必要な情報であると仮定し、それに対応する疑問詞の頻度と格の頻度の割合をしきい値として、質問に用いる疑問詞を選択した。これにより、用言に対応する質問候補である疑問詞と格のペアを取得する。したがって、この段階での出力結果は、「ある」という用言に対しては<どこ, ニ>、<誰, ニ>などのペア、「有名だ」という用言に対しては<何, デ>などとなる。

手順3 質問候補文を生成

手順2で取得した疑問詞と格のペアをもとに質問候補文を生成する。具体的にはノ格以外の場合は「疑問詞」+「格」+「正規化した用言」+「疑問形」を、ノ格の場合は「疑問詞」+「格」+「述語に係る名詞・格の繰り返し」+「正規化した用言」+「疑問形」をそれぞれ質問候補文として生成する。したがって手順2の出力結果より、入力発話例に対する出力結果は「どこに有りますか?」、「誰に有りますか?」、「何で有名ですか?」などとなる。

なお、「わかりました。」など、用言が含まれていても質問生成を行う必要がない入力発話も存在する。そこで、用言が一つも格要素を持たない場合は質問生成を行う必要がない発話であると判定し、質問候補の生成を行わないものとした。また、格フレームと対応辞書から推定した疑問詞の割合が少ない場合は質問生成を行う必要がないものと仮定し、それぞれの格の頻度に対してある割合以上の頻度を持つ疑問詞に対して質問候補を生成するものとした。例えば、しきい値として頻度の割合を10%と設定した場合、それぞれガ格の全体の頻度が1,000、<何, ガ>のペアの頻度が150、<どこ, ガ>のペアの頻度が50であったとすると、<

¹<http://nlp.ist.i.kyoto-u.ac.jp/index.php?KNP>

²<http://nlp.ist.i.kyoto-u.ac.jp/index.php?JUMAN++>

³<http://lotus.kuee.kyoto-u.ac.jp/nl-resource/jumanpp/jumanpp-manual-1.0.pdf>

何, ガ > のペアの頻度はガ格全体の 10%以上にあたるので質問候補を生成するが, < どこ, ガ > に関しては質問候補を生成しない。

3.2 質問選択手法

3.1 節で生成した質問候補に対して、焦点解析アルゴリズム [7] を用いて発話の焦点語の情報を用いることで、不要な質問候補の削除を行う。例えば、「大原からやはり近辺はバス停がちょっとないようです。はい。ですから、先ほどの、やっぱり、大原まで戻っていただけます。」という入力発話に対して、「何で戻りますか?」、「どこに戻りますか?」という質問候補を生成したとする。ここで、「どこに戻りますか?」という質問候補は、発話中に「大原」という場所の情報が出ていても関わらず、生成された二格ではなくマデ格の名詞であるために、< どこ, ニ > という疑問詞と格のペアから質問候補を生成してしまっている。このとき、入力発話の焦点語が「大原」なので対応する疑問詞は「どこ」となり、既に発話中に場所を表す情報が含まれていることから「どこ」を含む疑問詞を削除する。すなわち、今回の入力発話例から生成される質問候補は「何で戻りますか?」となる。

このようにして、入力発話中の焦点語の情報を用いて、既に入力発話中に含まれている情報を聞き返さないようにすることで質問候補の削除を行う。また、現発話から二つ前までの発話の焦点語の情報を保持することで、過去の発話の情報を用いた質問候補の削除も併せて検討した。

4 実験

4.1 評価セットの作成

提案した対話システムを評価するため、評価セットを作成した。対話データとしてはガイドと旅行者の間で京都観光に関してやり取りされた音声対話を書き起こした、京都観光案内対話コーパス [8] を用いた。このコーパスの一部に対して、3.1 節で提案した格フレームを用いる手法によって、もともとのコーパスに質問候補を加えたテキストを作成した。これは、対話コーパスに対して、アノテータに一から質問を考えてもらうことは負担がかかり、かつ質問を思いつかない場合の方が多いことが予想され、あらかじめ自動で質問候補を付与した方がカバレッジが高い評価セットを作成できると考えたためである。このとき、質問候補を付与するための頻度の割合のしきい値を 5% と設定した。

そして「対話コーパスに付与された質問候補を文脈を考慮した上で「必要な質問」、「聞いてもおかしく

2とアノテーションされた質問候補の数	31 個 (2.12%)
1とアノテーションされた質問候補の数	86 個 (5.87%)
0とアノテーションされた質問候補の数	1,347 個 (92.01%)
生成された質問候補の総数	1,464 個

表 1: 質問候補に対する評価セットの統計的な情報(「2」は「必要な質問」、「1」は「聞いてもおかしくはない質問」、「0」は「聞かない質問」をそれぞれ表す)

はない質問」、「聞かない質問」の三段階で評価してください」という指示のもと、3 人のアノテータにアノテーションを行ってもらった。3 人のうち、2 人以上が一致した評価をその質問候補の評価とし、3 人それぞれが異なる評価を行った場合は「聞いてもおかしくはない質問」と評価されたと解釈した。表 1 にアノテーションされた質問候補の統計的な情報を示す。「必要な質問」とアノテーションされた質問候補は全体の 2.12%にあたる 31 個のみとなったため、評価の際は「必要な質問」のみに対する評価と、「必要な質問」と「聞いてもおかしくはない質問」の両方を合わせたものに対する評価を用いた。

提案手法は、質問候補を過生成するため、Recall と比較して Precision が大幅に低下することが予想される。そこで単純に、正しい質問候補をどの程度カバーしているかを表す Recall と、システムが生成した質問候補のうちどれだけ正しいものが含まれているかを表す Precision を用いることとした。以降では便宜上、「必要な質問」とアノテーションされた質問を「2 の質問」、「聞いてもおかしくはない質問」とアノテーションされた質問を「1 の質問」と表現する。

4.2 生成した質問候補の精度評価

3.1 節で提案した、格フレーム内の疑問詞の頻度の割合のみを用いた場合と、3.2 節で述べた、入力発話の焦点語の情報を用いた場合、さらには二つ前の発話までの焦点語の情報を用いた場合の質問生成結果に対する評価をそれぞれ比較した結果を表 2 に示す。ここで、頻度の割合のしきい値は 10% として設定した。表 2 より、Precision と Recall のどちらも、焦点語を用いない場合の方が精度が良くなった。また、どの手法でも Precision の値がとて低くなった理由としては、正解の質問の数よりもとて多くの質問候補を生成したことが挙げられる。三つの手法のうち最も生成した質問候補の数が少なかった手法においても、「1 の質問」と「2 の質問」の数の合計が 117 個であったにもかかわらず、提案手法はその 6 倍よりも多い 733 個の質問候補を生成した。

提案手法	2 の Precision	1 と 2 の Precision	2 の Recall	1 と 2 の Recall
頻度の割合のみ	0.0224	0.0781	0.871	0.803
+ 焦点語の情報	0.0177	0.0707	0.581	0.615
+ 二つ前までの焦点語の情報	0.0139	0.0494	0.290	0.274

表 2: 生成した質問の精度評価

入力発話	嵯峨嵐山まで来まして、そこから歩いて、(焦点語: 嵯峨嵐山)
ラベル	質問候補
○	誰が歩きますか? →削除
	どこを歩きますか? →削除
	どこに歩きますか? →削除
○	何で歩きますか?
○	どこまで歩きますか? →削除
○	どこへ歩きますか? →削除

表 3: 焦点語の利用によって誤った例 (○: 必要な質問)

4.3 考察

焦点語を用いない場合に関して、生成した質問候補の数が最も多いために、三つの手法のうち最も Recall が高くなることは想定通りであったが、焦点語を用いた場合、予想に反して Precision が改善されなかった。これは生成した質問候補の数は少なくなったが、それ以上に正しい質問候補の生成が抑圧されてしまったと考えられる。焦点語の情報を用いたことで誤ってしまった例を表 3 に示す。入力発話の焦点語が「嵯峨嵐山」であるため、場所を聞く疑問詞「どこ」を含む質問候補を削除した結果、4 つ中 3 つの「必要な質問」を誤って削除してしまい、Precision が低下した。このように、入力発話の焦点語の情報のみでは質問候補の削除を誤ってしまうケースが多発した。

5 おわりに

本稿では人手で複雑なルールを設計せず、また大量の学習データを必要としないような、格フレームを用いた質問候補の生成手法を提案した。実験では評価セットを作成し、その評価セットを用いて提案システムを評価したところ、必要な質問に対する Precision は 0.0224、Recall は 0.871 となり、Precision は特に低い値となった。また、Precision の改善のために焦点語の情報を用いたところ、必要な質問に対する Precision は 0.0177、Recall は 0.581 となり、焦点語の情報を用いない場合と比べて値が低くなった。提案手法により、ある程度のカバレッジで必要な質問を生成可能であることがわかったので、今後は不要な質問を削減する手法を検討する必要がある。

謝辞

本研究の一部は JST さきがけ (JPMJPR165B) の支援を受けた。

参考文献

- [1] Koichiro Yoshino, Yu Suzuki, and Satoshi Nakamura. Information navigation system with discovering user interests. In *Proceedings of the 18th Annual SIGDial Meeting on Discourse and Dialogue*, pp. 356–359. Association for Computational Linguistics, 2017.
- [2] 下岡和也, 徳久良子, 吉村貴克, 星野博之, 渡部生聖. 音声対話ロボットのための傾聴システムの開発. *自然言語処理*, Vol. 24, No. 1, pp. 3–47, 2017.
- [3] Rafael E. Banchs and Seokhwan Kim. Spoken dialogue system for restaurant recommendation and reservation. In *INTER_SPEECH-2014*, pp. 1488–1489, 2014.
- [4] Xingwu Sun, Jing Liu, Yajuan Lyu, Wei He, Yanjun Ma, and Shi Wang. Answer-focused and position-aware neural question generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 3930–3939. Association for Computational Linguistics, 2018.
- [5] Yansen Wang, Chenyi Liu, Minlie Huang, and Liqiang Nie. Learning to ask questions in open-domain conversational systems with typed decoders. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2193–2203. Association for Computational Linguistics, 2018.
- [6] Daisuke Kawahara and Sadao Kurohashi. Case frame compilation from the web using high-performance computing. In *Proceedings of the 5th International Conference on Language Resources and Evaluation*, pp. 1344–1347, 2006.
- [7] Koichiro Yoshino and Tatsuya Kawahara. Information navigation system based on pomdp that tracks user focus. In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pp. 32–40. Association for Computational Linguistics, 2014.
- [8] 翠輝久, 大竹清敬, 堀智織, 柏岡秀紀, 中村哲. 京都観光案内対話コーパスにおける対話行為タグの設計と分析. *情報処理学会研究報告 音声言語情報処理 (SLP)*, Vol. 2009, No. 10, pp. 39–44, jan 2009.