

話題展開構造を用いる擬似会話生成システムの提案と実装

Proposal and Implementation of pseudo-conversational generation system using a topic deployment structure

丹下 雄太*¹ 大本 義正*¹ 西田 豊明*¹
Yuta Tange Yoshimasa Ohmoto Toyoaki Nishida

*¹京都大学 情報学研究科 知能情報学専攻

Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University

Originally, humans consider the configuration composed of topics when a conversation. Three distinctive structures, which are the smooth connection between the topic, the belt-like structure, and the loop structure of composed of topics, was observed in human conversations. In this study, we proposed how to reproduce these three structure in generated dialog by on-demand generating pseudo-conversation-dialog system, and we implement and evaluated the system.

1. はじめに

近年、対話型の秘書アプリケーションが人間の音声命令に従ったり、ヒューマノイドロボットが人間に対して接客を行うなど、システムと人間が言語によるコミュニケーションを行うことが社会に必要とされてきており、それに伴い言語による対話インターフェースの自然さの重要性が提唱されるようになってきた。

一方で TwitterBot のような、会話ボット (人工無脳) と人間との会話が一般的に認知されるようになってきたが、現状の会話ボットの会話生成手法は自然な対話インターフェースとして用いることが可能なレベルには至っていない。

現在登場している会話ボットには、単語によるパターンマッチングや、前後の発話のペアの学習等、単純な手法を用いているものが多い。

また Peedy や SHRDLU[3] 等、対話を通じてなんらかの作業を行う対話エージェントには、作業領域等を状態として記述し、発話をエッジとした状態遷移図が内包されている。これらの対話エージェントが意図しているのは、状況に応じた自然な発話の実現である。

本来、人間は会話をを行いながら、談話単位で内容を抽象化し、その流れを記憶しており、その構成を意識しながら会話を進めると考えられる。現在の対話エージェントにこの構成に着目したものは見受けられなかった。本稿では、この構成のことを話題展開と呼ぶ。

対話エージェントが行う会話が人間に自然だと判断されないのは、話題展開の自然さの実現が十分でないことが原因の一つであると考えられる。

よって、「自然な話題展開を実現する対話エージェントの実装」を本研究の最終目標とする。この最終目標達成によって、対話エージェントが行う会話の自然さの向上が期待できる。

自然な話題展開を実現する対話エージェントの実装の為に、

発話同士の自然な繋がり 前後の発話同士が自然に繋がっている構造。

ベルト状の話題展開構造 任意の時点の発話が過去の話題の連なりの影響を受けて提示されることによって形成される構造。

ループ状の話題展開構造 話題が明確に逸れた時に、過去の話題に戻ることで形成される構造。

が必要であると考えられる。これは下に行くほど大きな構造を示す。

自然な話題展開に必要な要因として、まず第一に挙げられるのは「発話同士の自然な繋がり」であろう。これだけで自然

な話題展開が表現できるのであれば、現在の会話ボットや対話エージェントによって自然な話題展開を実現することができていないはずであろう。

よって本研究では、発話同士の自然な繋がりに加えて、より大きな構造、即ち上記のベルト状の話題展開構造、ループ状の話題展開構造が自然な話題展開の実現に重要な役割を担っていることを示すこととする。

これを実現するため、先で述べた 3 つの構造を、本研究で扱う自然な話題展開実現の会話生成アルゴリズムの中に組み込むこととする。しかしベルト状の話題展開構造、ループ状の話題展開構造は非常に抽象的な概念でしか定義することができないので、実装が難しい。

形式意味論における意味の関係理論の考え方では、言語の意味は状況や信念を変化させる機能であるとしている。この考え方を採用した談話管理理論においては、対話における発話を話者と聴者に関する知識からなるデータベースを変化させる写像としてモデル化している。また、対話を通じてなんらかの作業を行う対話エージェント (Peedy, SHRDLU[3]) の多くには、作業領域等を状態として記述し、発話をエッジとした状態遷移図が内包されている。

本研究ではこれらの考え方を参考に、発話を状態遷移におけるエッジと捉え、既存の会話から発話の前状態と後状態を推測し、発話単位で分割、状態の抽象的記述を行うことで、発話を状態遷移モデルとしてデータベース化し、それら状態遷移モデルを統合することで (これによって構築された構造を本稿では状態遷移構造と呼ぶ) 会話文を生成するようなシステムを構築する。

この状態遷移モデルの統合の仕組みに先に記した 3 つの構造を構成するための実装を組み込み、システムの妥当性検証のため、アンケートによる評価を行う。

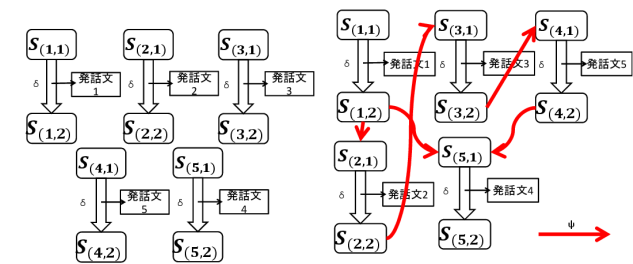
2. 擬似会話生成システム

本研究の本論である擬似会話生成の手法について述べる。

人間同士の会話においては、発話の前に位置する発話の条件と後に位置する発話の条件が異なる場合があり、また発話というものは前の発話を踏まえて行うものであるから、1 発話に 1 つの特徴ベクトルを対応付けることでは自然な発話の繋がりや過去の話題への回帰を行う際の橋渡しの役割を実現することができない。

従って、発話を前状態から後状態への遷移を行うエッジと捉えた発話の状態遷移モデルを用い、それらを統合することで擬似会話を生成することとした。

連絡先: 丹下雄太, 京都大学大学院情報学研究科 知能情報学専攻 西田研究室, 京都府京都市左京区吉田上阿達町 7-7 カサローゼ吉田 205, 09098724351, nero21jp@gmail.com



(a) 複数の状態遷移モデルによる断続的な有限オートマトン (b) 状態遷移構造

図 1: 基本システム

2.1 発話の状態遷移モデル

例 1 前後の条件相違が生じる発話

ウイスキーが美味しいのは良く分かっているんだけど、
合うつまみがなかなか見つからなくてねえ

例 1 では「ウイスキーが美味しい」という主旨の発話が前に来ることが条件と考えられるが、後には「ウイスキーとつまみ」に関する発話来ることが条件だと考えられる。従って、発話と 1 対 1 対応の表現では発話同士を繋げることが困難である。これを解決する為、発話を状態から状態へのエッジであると見なし、解決を図る。これは先に述べた談話管理理論や Peedy,SHRDLU に用いられている方法を参考にしている。

説明の為、以下の有限オートマトンを考える。

Γ 出力文字セット S 状態集合

δ 状態遷移及び出力関数, $\delta: S \rightarrow S$

状態 S_1 と状態 S_2 が存在し、この 2 状態間に S_1 から S_2 へのエッジ $\delta(S_1) = S_2$ が存在するとする。このとき、このエッジにより現状態が状態 S_1 であれば状態 S_2 への遷移が可能となり、この遷移により発話文が出力される。これを複数集めることで、断続的な有限オートマトンを形成する。

複数の発話の状態遷移モデル

$\Gamma = \{ ' 発話文 1', ' 発話文 2', \dots, ' 発話文 n' \}$

$S = \{ S_{1,1}, S_{1,2}, S_{2,1}, S_{2,2}, \dots, S_{n,1}, S_{n,2} \}$

$\delta(S_{i,1}) = (S_{i,2}, ' 発話文 i') \quad i = (1, 2, \dots, n)$

以降、 $S_{i,1}$ をエッジの前状態、 $S_{i,2}$ をエッジの後状態と呼ぶ。

2.2 状態の記述

ここでの状態とはある時点でのあらゆる物体、環境を瞬間的に切り取ったもののことであるが、今回は既存の会話からエッジの前状態及び後状態を推測することとする。

従って前後の発話及びその発話自体から推測される条件のみを特徴量として状態の記述に用いることとする。

2.2.1 多重集合による状態の記述

前節の状態記述は概念的なものであるため、前状態と後状態を演算可能にする為の準備としてこれを多重集合として表す。

今後、紛れの恐れのない限り、状態とは、その状態を表す多重集合を表すものとする。

状態 X における要素 a の個数を $m_X(a)$ 、もしくは X における a の重複度と呼ぶ。状態 X の要素の種類を表す集合を \tilde{X} 、もしくは X の台集合と呼ぶ。

2.3 近似状態の発見による状態遷移構造の構築

2.3.1 近似状態

既存の発話を分割し、それぞれの発話が 4.2 節で示した状態遷移モデルとして表されたとする。

ここで、これらの状態遷移モデルを何らかの方法で統合し、連続的にエッジによる状態遷移が行われれば、連続的に発話文が出力され、会話文生成が可能であるはずである。

従って、ある関数 ψ が存在し、関数 ψ が以下のような形を成していれば、状態遷移モデルは統合される。

$$\psi: S \rightarrow S, \psi(S_X, \theta) = (S_Y, \eta)$$

以降、この関数 ψ を近似状態移動関数と呼び、 $\psi(S_X) = S_Y$ となるような状態 S_Y を前状態 S_X の近似状態と呼ぶ。 θ, η は近似状態の決定に必要ななんらかの変数である。また、統合後の構造を状態遷移構造と呼ぶこととする。

状態遷移構造を遷移し続けることにより、連続的に発話出力され、擬似会話生成が成される。

近似状態の決定方法によって近似状態移動関数及び状態遷移構造が変わり、生成される擬似会話文の構造も大きく変わる。

以降は、この近似状態移動関数をいくつか挙げ、それぞれによって期待される会話文構造を述べる。

2.3.2 状態間の積

ここで、状態間の積を定義する。状態間の積に関しては以下に従う。

$$X \cap Y = \{ \tilde{X} \cap \tilde{Y}, m_{X \cap Y} \}$$

$$m_{X \cap Y}(a) = \min\{m_X(a), m_Y(a)\}$$

以降、状態 X と状態 Y の積を、 X と Y の積状態と記述する。

両者とも*が存在する要素を含まない状態同士の積においては、今後前状態、後状態、積状態共区別せず、上記の定義に従い積状態を計算する。積の計算において片方*が存在する要素を含む状態であるときのみ、以下に記述する法則に従う。

尚、*を含む特徴量については、積をとる前状態が*で区切られた特徴量の全てを含む場合においてのみ、積状態に追加する。

$$\begin{aligned} \text{feature1*feature2} \in Y \wedge \text{feature1} \in X \wedge \text{feature2} \in X \\ \rightarrow \text{feature1*feature2} \in (X \cap Y) \end{aligned}$$

2.3.3 前後状態比較による発話同士の自然な繋がりの再現

2.3.2 節において説明した近似状態を前状態と後状態の積状態の要素の数により決定する方法を説明する。状態 X の要素の数を $|X|$ で表し、以下で定義する。

$$|X| = \sum_{a \in X} m_X(a)$$

積状態の要素の数が本質的に表しているのは条件の一致の度合いである。よって積状態の要素の数が多いほど、状態と状態は類似しているということになる。

まず、ある後状態に対して、全ての前状態との積状態を計算し、要素数が最も高い物をその後状態の近似状態とする。

これを全ての後状態に対して行うことで、全ての後状態には近似状態が存在し、エッジの後状態から別のエッジの前状態への移動が可能となり、連続的な遷移が可能になる。これによって発話同士の自然な繋がりを再現する。

この近似状態の発見方法を以後、前後状態比較と呼ぶこととする。

$$S = \{ S_{1,1}, S_{2,2}, S_{2,1}, S_{2,2}, \dots, S_{n,1}, S_{n,2} \}$$

$$\delta(S_{i,1}) = (S_{i,2}, ' 発話文 i') \quad i = (1, 2, \dots, n)$$

$$\psi(S_{x,2}) = S_{l,1} \quad (l \mid |S_{x,2} \cap S_{l,1}| = \max_{i=1, \dots, n} |S_{x,2} \cap S_{i,1}|)$$

$$(x = 1, \dots, n)$$

2.3.4 状態遷移による会話生成における既遷移状態の削除

同じ発話文が繰り返し出力される事を防ぐため、一度遷移・移動した状態は近似状態の候補から消す。つまり、あるエッジ E の後状態 $S_{E,2}$ から近似状態として別のエッジ E の前状態に移動した場合、上記のオートマトンにおいてエッジ E の前状態 $S_{E,1}$ と後状態 $S_{E,2}$ を状態集合 S から削除する。これを近似状態移動が行われる度に動的に行う。以降における他の近似状態の決定方法に基づく会話生成でもこれを同様に行う。

2.3.5 問題点、間接的な話題展開表現

前節までの手法により、任意のエッジの前状態から状態遷移と近似状態への移動を繰り返すことで擬似会話生成が行われるが、これは刹那的な状態の類似度によって状態遷移構造を繋げているに過ぎない。即ち、ある一つのエッジの前状態と後状態の一致度が低いようなエッジを考えると、こういったエッジを何度か通ることで、現状態が初期状態と全く異なった状態である可能性が高い。このようにして生成された会話は、話が逸れ続けるような会話であると考えられる。従って、話の逸れを防ぐ為、より大きな構造を会話文中に組み込む必要がある。

前後状態比較で前状態と後状態の積状態を用いたが、この積状態は発話同士繋がりを抽象的に表した概念であると考えられる。

ここで、状態遷移構造において遷移を繰り返した時、後状態からその近似状態へ近似状態移動が行われるが、この2状態の積状態を近似状態移動が起こる度に蓄積したとすると、この蓄積した複数の積状態は、今までの会話を抽象的な形で表している。

これを利用し、先に述べたベルト型の話題展開構造、ループ型の話題展開構造を会話文中に再現することで、問題の解決を図る。状態の蓄積に関しては計算機科学における入力制限型デッキという概念を用い、この2構造を再現する近似状態移動の手法として話題記憶モデル、話題回帰モデルを説明する。

2.4 入力制限型デッキ

計算機科学におけるデータ構造に、入力制限型デッキ [5] というものが存在する。入力制限型デッキとは、要素の取り出しは両端で可能だが、追加は一方からしか行うことのできないデッキのことである。この概念を利用し、過去状態を順に蓄積し、過去の遷移ルートの記憶を可能にする。

追加可能な端を 端、もう一方の端を 端と呼ぶことにする。エッジによる遷移によってある後状態に遷移した場合、その後状態を 端から追加し、ある後状態から近似状態への移動が生じた場合、 端に最も近い状態を参照し、その状態と現状態との積状態を、 端から状態の一つ取り出した後に追加する。また、記憶容量を超えて追加する場合は、 端から集合を取り出す。

以下このデータ構造をデッキと呼ぶ。これは、古い記憶から抜けていき、新しいものから思い出す人間の記憶を再現している。

以降、デッキを用いて近似状態を決定、移動することで、ベルト型の話題展開構造、ループ型の話題展開構造を再現する方法として、話題記憶モデル、話題回帰モデルを説明する。

2.5 話題記憶モデルによるベルト型の話題展開構造の再現

ある特定の時点におけるデッキ内の全ての状態の直和集合を考える。

この直和集合は多重集合であるため、デッキ内に存在している多重集合のうち、共通して持っている要素が存在すれば、その分話題におけるその要素の重複度は高くなっているはずである。従って、この直和集合を前述の前後状態比較における現状態の代わりとして、直和集合との積状態の計算を行い、要素数が最も多い前状態を近似状態として移動すれば、必然的に重複度の高い特徴を含む状態が優先的に選出される。

常に、重複度の高い要素が主軸となって状態遷移を行うことで、会話の軸を擬似的に表現する。話題記憶モデルのアルゴリズムは以下ようになる。

1. 初期状態をデッキに追加する。
2. エッジによる遷移を行う。(発話文出力)
3. 現状態をデッキに追加する。

4. デッキ内の全ての状態の直和集合と、候補として残る全ての前状態との積集合を計算し、積集合の要素数の最も高い前状態を近似状態とし、積集合をデッキに追加し、近似状態へ移動する。(2へ)

この手法を以後話題記憶モデルと呼ぶ。

2.6 話題回帰モデルによるループ型の話題展開構造の再現

これまでの議論により、前後状態比較、話題記憶モデルの2種類の近似状態の発見方法を提案したが、これらいずれかの方法によって状態遷移を行うことで、デッキ内に過去に遷移した状態が蓄積されていく。なんらかのタイミングで、デッキ内に蓄積された状態に現状態を戻す手法を説明する。

デッキ内が以下の様になっているとき、状態 X_n を 端から取り出す場合、状態 X_n より 端側(ここでは左)にある状態 X_1 から X_n も同時に取り出されるとする。以降、このように 端から状態を取り出すことをポップと呼ぶ。

$$\alpha \text{端} (X_1, X_2, \dots, X_n, X_{n+1}, \dots, X_m) \beta \text{端}$$

定期的にポップを行うことで、現状態から過去状態へ戻し、話が逸れつづけることを防ぐと共に、遷移のループや節となる状態を作りだし、特定の状態から離れすぎないようにする。

本研究では、エッジを介してポップを行い、これを話題回帰モデルとして説明する。ポップを行う際、候補集合に存在する全てのエッジとデッキに存在する全ての状態の組に対し、以下の計算を行う。

$$C(i, X_n) = \text{エッジ } i \text{ の前状態と現状態の積状態の要素の数}$$

+ エッジ i の後状態とデッキ中の状態 X_n の積状態の要素の数

$$= |S_{i,1} \cap X_1| + |S_{i,2} \cap X_n|$$

$C(A_i, X_m)$ の値が最も大きい場合、現状態からエッジ A_i の前状態へ移動し、エッジ A_i による状態遷移を行った後、状態 X_m に移動する。その後、 X_{m-1} をデッキからポップする。これにより、現状態を以前デッキに追加した状態にエッジを介して滑らかに戻ることが可能となる。この手法を以後話題回帰モデルと呼ぶ。

3. 評価実験

先の議論に基づいたオンデマンド会話生成システムを構築し、システムの妥当性検証の為、出力した会話文に対してアンケートによる評価を行った。

3.1 実験内容

今回、「2ちゃんねる掲示板」の「お酒・BAR」のジャンルより、スレッド5つから200発話から、データベースを作成した。

ランダム、前後状態比較、話題記憶モデル、話題回帰モデルの4つの手法により、データベースから20~30発話で構成される会話文をそれぞれ10個生成し、4つの手法で生成された会話文をそれぞれ1つずつ被験者に見せ、アンケート評価を9人の被験者に対して行った。また、デッキ容量は10状態で行った。

尚、話題回帰に関しては、基本的に前後状態比較によって会話文を生成、人間が「話の逸れ」を監視し、「話の逸れ」が感じられた時のみ回帰命令を出し、回帰を行うこととした。

実験の出力例は以下の通りである。

例2 出力例:前後状態比較

ウイスキーストレートで飲んだら喉痛めないですか？

喉よりも胃腸へのダメージを心配した方がいい

酒で危険なのは内臓より脳萎縮

俺は考えすぎて人生損してるからちょっと脳萎縮するくらいが丁度いい

俺もさんざん呑み屋通いしてきたが、今じゃ病院通いの日々。

...

また、手法同士の評価の差が有意なものであるかをみる為、得られた結果に対し、有意水準 $\alpha = 0.05$ のマン・ホイットニーの U 検定を行い、検定によって得られた P 値が 0.05 以下の値を示した時に有意差、0.1 以下の値を示した時に有意傾向があるものと判断した。

3.2 結果と考察

システム中の近似状態の発見方法はそれぞれ、前後状態比較は発話同士の自然な繋がり、話題記憶モデルはベルト状の話題展開構造、話題回帰モデルはループ状の話題展開構造を構成するアルゴリズムであり、これらの構造が話題展開の自然さにどの様に影響するか、評価結果より考察する。アンケートによる評価結果を表 1 に示す。

質問	ラ	前	記	回
話題展開の自然さ	0.225	0.644	0.578	0.822
話のまとまり	0.333	0.600	0.600	0.756
全体の人間らしさ	0.444	0.725	0.733	0.756
チャット, 前後文の繋がり	0.347	0.843	0.655	0.77
音声会話, 前後文の繋がり	0.285	0.716	0.472	0.659

表 1: 評価結果, 平均点数 (満点を 1.0 としている)

3.3 前後状態比較

話題展開の自然さにおいて、ランダム生成による会話文は点数平均 0.225(満点 1) であったのに対し、前後状態比較は 0.664 と明らかに評価が高く、有意差も出ていた。このことから、「発話同士の自然な繋がり」は話題展開の自然さにおいて重要な要因であると分かる。

3.4 話題記憶モデル

話題展開の自然さにおいて、話題記憶モデルは 0.578 と明らかに評価が高く、有意差も見られた。

しかし、話題記憶モデルは「文と文の繋がり」においてランダム生成と有意差が見られないほど著しく評価が低かった。一方で、話のまとまり、全体の人間らしさの評価は高かった。話題記憶モデルは、デッキ内の全ての状態の直和とのマッチングを測るものであるため、現状態の要素による影響が低くなり、全体の話のまとまりの評価は高くなる一方で、文同士の繋がり疎かになるパターンが存在し、これが影響したものと考えられる。

「文と文の繋がり」の評価がランダム生成との有意差が示されない程低かったにも関わらず、話題記憶モデルが「話題展開の自然さ」の評価でランダム生成に比べて評価が高い有意差が示され、前後状態比較との有意差が示されなかったことから、話題記憶モデルによって構成される「ベルト状の話題展開構造」が「話題展開の自然さ」の重要な要因となっていると言えるだろう。

3.5 話題回帰モデル

話題展開の自然さにおいて、話題回帰モデルは 0.822 と最も評価が高く、有意差も見られた。

話題回帰モデルに関しては、そもそも今回の評価実験において、話題回帰モデルは回帰時以外は前後状態比較によって発話を繋げており、本来話題回帰モデルが構成するループ状の話題展開構造が「話題展開の自然さ」の要因でないならば、前後状態比較とさほど変わらない評価結果となるはずだが、「話題展開の自然さ」において、前後状態比較と話題回帰モデルは、話題回帰モデルの方が評価が高い有意傾向が見られていた。

このことから、話題記憶モデルによって構成される「ループ状の話題展開構造」が「話題展開の自然さ」の重要な要因となっていることは判断できる。

4. 総合考察

自然な話題展開を実現するシステムの実装方法の提案、実装、評価を行い、評価結果より、発話同士の繋がりに加えて、ベルト状の話題展開構造、ループ状の話題展開構造が自然な話題展開の実現に重要な役割を担っていると分かった。

また多くの質問に関して、提案した 3 手法は評価が高く、疑似会話生成の手法としても効果的であったと考えられる。

さらに付加的な知見として、「会話文全体の人間らしさ」における評価から、ある程度前後の繋がりを疎かにしても、話題のまとまりを意識した話題記憶モデルや話題回帰モデルの方が前後のみを比較する前後状態比較よりも「人間らしい会話文」の生成には効果的であることが分かった。

最終的に、話題記憶、話題回帰モデルの統合により、人間らしい話題展開を実現するシステムは実現可能だと予測される。

5. 結論

本研究では「発話同士の繋がり」に加えて、ベルト状の話題展開構造、ループ状の話題展開構造が自然な話題展開の実現に重要な役割を担っていることを示すことを本目的とし、自然な話題展開に必要な要因として発話同士の自然な繋がり、ベルト状の話題展開構造、ループ状の話題展開構造の 3 つを会話文中で構成するため、発話を状態遷移のエッジと捉え、既存の会話から発話の前状態、後状態を推測、発話単位で分割、状態の抽象的記述を行うことで、発話を状態遷移モデルとしてデータベース化し、状態遷移モデル統合の仕組みにおいて、積状態の要素数から発話同士の自然な繋がりを構築し、入力制限型デッキの概念を反映したデータ構造を用い、ベルト状の話題展開構造及びループ状の話題展開構造を構築する手法を提案、実装、評価を行った。

評価結果より、「発話同士の繋がり、話題記憶モデル、話題回帰モデルは三者とも、自然な話題展開の実現に重要な役割を担っている」と判断し、本研究の本目的は達成は成されたとする。

また、以上の研究プロセスより、自然な話題展開の実現方法の提案と、実装して実験的に確かめることもできた。

本研究によって、自然な話題展開を実現する対話エージェントの実現に対し、新しい方向性を示したという点において、これは進歩と言えるだろう。

参考文献

- [1] 服部峻: Web 知識を用いた時空間依存な対話システムの試作 (「Web インテリジェンス」及び一般), 電子情報通信学会技術研究報告. AI, 人工知能と知識処理, Vol. 110, No. 105, pp. 13-18 (2010).
- [2] 伝康晴, 長尾眞: 談話管理理論を用いた対話処理: 名詞句の解析と生成について, 人工知能学会誌, Vol. 6, No. 6, pp. 872-880 (1991).
- [3] 徳永健伸: ロボットとの会話人工知能からのアプローチ, 情報処理, Vol. 44, No. 12, pp. 1247-1252 (2003).
- [4] 鬼頭隆, 増塩智宏, 安村禎明, 新田克己: 状態遷移モデルを用いた交渉会話のシミュレーション, 情報処理学会研究報告. ICS, [知能と複雑系], Vol. 99, No. 63, pp. 17-24 (1999).
- [5] Donald, K.: The Art of Computer Programming, Volume 1: Fundamental Algorithms, Third Edition. Addison-Wesley, ISBN 0-201-89683-4. Section 2.2.1: Stacks, Queues, and Deques, pp. 238-243 (1997).