

E-009

モジュラリティの高い対話制御開発環境の試作

Making prototype development environments of dialogue control with high modularity

繁田 佳宏[†] 池ヶ谷 有紀[†] 野口 靖浩[†] 小暮 悟[†] 伊藤 敏彦[†]
 Yoshihiro Shigeta Yuki Ikegaya Yasuhiro Noguchi Satoru Kogure Toshihiko Itoh
 小西 達裕[†] 近藤 真[†] 伊東 幸宏[†]
 Tatsuhiro Konishi Makoto Kondo Yukihiro Itoh

1. はじめに

すでに存在する多くの対話システムは、システムの流用性やドメイン独立性が低く、別のドメインに適用しようとした場合、そのコストは低いものではない。その一方で、ドメイン独立性を高めた対話システム開発環境についての研究もいくつか存在する。鈴木ら[1]はデータベース検索、予約のドメインに特化した上で音声対話システム開発のコスト削減を行っている。

我々もこれまでに、より広い範囲でのドメイン適用が可能な対話システムを構築してきた[2]。しかし、[2]のシステムは、システムの評価及びその対話記述能力が不十分であったため、今回、実現する対話の流れを記述する形式を再整理し、対話記述能力の向上と、記述コスト削減を目指した。

本研究では、対話システムをより簡単に作成できる開発環境を整えるための前準備として、対話制御部に注目し、あらかじめ用意してある対話制御用部品の組み合わせを使って対話をスクリプトで記述することで、様々な対話を簡単に実現できるシステムを構築した。

2. モジュラリティの高い対話制御部品

対話を1ターンの粒度で見ると、対話は、ユーザからの入力とシステムの内部状態に応じて起動される一つ、もしくは幾つかのルールによって制御できると考えられる。この粒度で定まるルールを本稿では「プリミティブタスク」と定義する。もしも、完全なプリミティブタスクのセットが定義できるのであれば、そのセットを組み合わせることで、すべての対話が実現可能となるはずである。しかし、プリミティブタスクのみで対話を記述しようとする、その粒度が小さく対話システム設計者の作業量が多くなり、そのまま利用することは困難である。そこでいくつかの対話例を分析し、それらの対話で頻繁に出てくるプリミティブタスクの組み合わせを、7つの「サブタスク」として定義している[2]。以下の節で、それぞれのサブタスクについて述べる。

2.1 あいさつ

このサブタスクは、あいさつ文をシステムが発話し、次のサブタスクへ遷移する。

2.2 確認

このサブタスクではユーザに確認を取る。確認項目を尋ね、ユーザはその確認項目に対して yes/no で答えることができる。yes/no の結果により遷移先サブタスクが決定す

る。

2.3 情報を伝える処理

このサブタスクはユーザに情報を与えるためのサブタスクであり、システムが何らかの情報を伝えたいときに実行される。伝達情報を一文ずつ伝達し、その都度ユーザから了解を得る。またユーザからの了解が一定時間ない時は、了解したものとして、処理を行う。

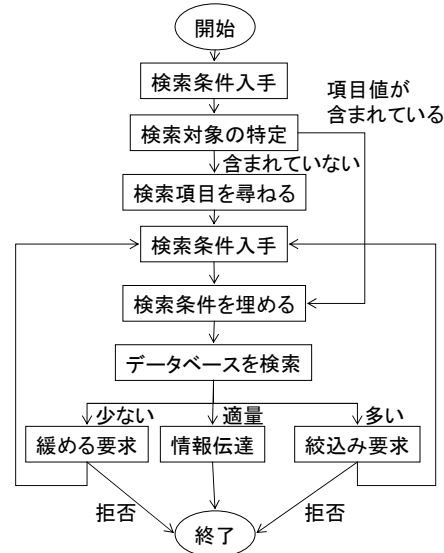


図1: データベース検索

2.4 データベース検索

このサブタスクは、文献検索、ホテル検索などの検索対話で用いられる。このサブタスクの基本動作を図1に示す。ユーザからの検索要求をトリガとして、サブタスクが開始し、図1中の3つの終了条件のいずれかが成立すれば終了する。また質問内容やデータベースに関する記述などについては、ドメイン知識として記述する。

2.5 質問受け付け

このサブタスクは、ユーザからの割り込み的な質問に回答することを想定したサブタスクである。基本的にはデータベース検索と同様の働きをするが、答えることができない質問もこのサブタスクで受け付け、答えることができない質問に対しては、答えることができないことをユーザに伝えて、対話が途切れないようにする為の遷移も含まれている。

[†] 静岡大学情報学部

[‡] 北海道大学大学院情報科学研究科

