INFORMATICS INSTITUTE OF TECHNOLOGY

Informatics Institute of Technology
In Collaboration With
University of Westminster

Cuneiform

A Visual Programming Environment for Conversational Interfaces

A dissertation by
Savidu Dias

Supervised by
Dr. Thilak Chaminda

Submitted in partial fulfilment of the requirements for the

BSc (Hons) Software Engineering

Department of Computing

May 2018

Abstract

Conversation is the most natural way humans interact with each other. Therefore, many researchers have directed their efforts towards HCI systems that interact with their users through natural language. Such systems are known as conversational interfaces.

There has been a rapid increase in the development of systems utilizing conversational interfaces. Many such systems provide support for third party developers to create applications. Since the concept of conversational interfaces is new, each system has its own approach to application development, and may have their own language syntax. As a result, development of applications for multiple systems becomes a tedious task for developers.

To address this problem, this project creates a federation between multiple conversational platforms, and provides a unified development environment. Its key contribution is the introduction of a programming language for conversation development. This is accompanied by a developer tool to visually design the conversation flow. This platform is named **Cuneiform**, and is a novel approach to the development of conversational interfaces. Cuneiform can be used to develop applications for existing VA and chatbot platforms such as Amazon Alexa, Facebook Messenger, or novel applications for custom systems interacting with users through conversation.

The main components of Cuneiform are the Responder, Classifier, and the Dialogue Manager. These components are implemented as three individual, interconnected processes. The Dialogue Manager contains the core of the system.

The Development Environment enables developers to create conversational applications through the utilization of a Conversation Graph, which is represented by the Cuneiform programming language. The Cuneiform script written by the developer is interpreted by an interpreter within the Dialogue Manager.

After the project was designed and implemented, it was evaluated by three target groups: conversational application developers, non-conversational application developers, and domain/industrial experts. The complexity, feasibility, and usability of the platform were evaluated by the first two target groups. Evaluation results from non-conversational application developers showed that they claim the platform has a medium level of complexity, while conversational application developers have stated that the system is easy to use. Both groups however agreed that the platform is feasible, and had a positive outlook on the user interface and user experience of the platform.

The overall impressions of the platforms from domain and industrial experts were positive. Many pointed out the wide-ranging applicability of the project. A few domain experts suggested better approaches for dialogue management which could be taken as future enhancements.