Conversational H.C. Andersen
First Prototype Description

Niels Ole Bernsen, Marcela Charfuelàn, Andrea Corradini, Laila Dybkjær, Thomas Hansen, Svend Kiilerich, Mykola Kolodnytsky, Dmytro Kupkin and Manish Mehta

NISLab, University of Southern Denmark, Odense

Introduction

This paper describes the running first prototype (PT1) of the NICE Hans Christian Andersen system. NICE stands for Natural Interactive Communication for Edutainment (http://www.niceproject.com/). In this EU project (2002-2005), we aim to demonstrate domain-oriented conversation, including 2D gesture input, with life-like animated fairytale author Hans Christian Andersen (HCA). By contrast with task-oriented spoken dialogue [1], domain-oriented conversation has no task constraints. The user can address, in any order, any topic within HCA's knowledge domains, using spontaneous speech and mixed-initiative dialogue. In PT1, the domains are: HCA's works, his life, his physical presence in his study, the user, and HCA's role as "gate-keeper" for access to the fairytale world which is not described here. In addition, HCA has a 'meta' domain to be able to handle meta-communication during conversation. HCA reacts emotionally to the user's input, e.g. by getting angry or sad due to what the user says, or happy if the user likes to talk about his fairytales. The HCA system is *not* an information system. It attains its educational goal by providing correct factual information, both visually and orally, but an equally important goal is to entertain through human-like conversation, to make the target users of 10-18 years old kids and teenagers pleased by having met someone of, and from, a different age who is much more like themselves than expected.

Below, we present the HCA system architecture, focusing on general architecture and information flow, as well as NISLab's natural language understanding, character modelling, and response generation modules.

General architecture

The HCA system's event driven, modular, asynchronous architecture is shown in Figure 1. In addition to the modules explained in more detail below, modules are (provided by): speech recogniser (Scansoft, not in PT1); gesture recognition (freeware); gesture interpretation, input fusion (LIMSI, no semantic fusion in PT1); speech synthesis (Scansoft), including time calculation for animation tags; and animation, including character animation and virtual world simulation (Liquid Media). The modules communicate via a central message broker, publicly available from KTH at http://-www.speech.kth.se/broker. The broker is a server which routes function calls, results and error codes between modules. The Transmission Control Protocol (TCP) is used for communication. The broker coordinates input and output events by time-stamping all module messages and associating them to a certain conversation turn. The behaviour of the broker is controlled by message-passing rules, specifying how to react when receiving a message of a certain type from one of the modules.

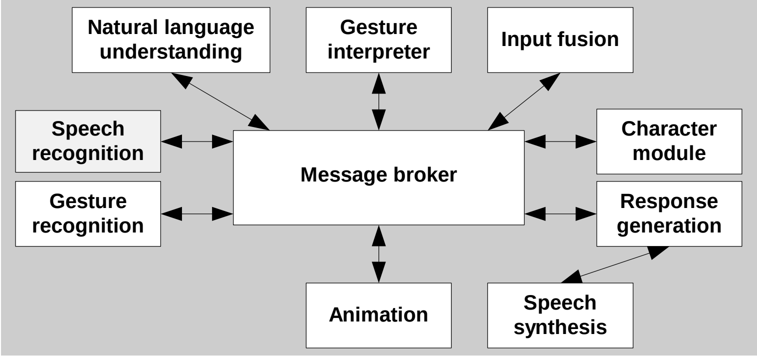

Figure 1. General NICE HCA system architecture.

In terms of information flow, the speech recogniser sends an n-best set of hypotheses (PT2) to natural language understanding which sends a 1-best hypothesis to input fusion. Similarly, the gesture recogniser sends an n-best hypothesis set to the gesture interpreter which consults the animation module as to which object the user may have indicated. In PT1, the input fusion module simply forwards an n-best list of pairs of (recognised pointable object + gesture confidence score) from the gesture interpreter and/or a 1-best natural language understanding output to the character module which takes care of input fusion, when required. The character module sends a coordinated verbal/non-verbal output specification to the response generator which splits the output into synchronised text-to-speech and animation. Synchronisation is handled by the animation module. For comparison, see, e.g., the architectures in [2].

Language understanding, character module, response generation

The natural language understanding (NLU) module manager (Figure 2) manages internal NLU communication. Each domain has a set of keyphrases. The keyphrase spotter spots phrases in the user utterance and converts them into syntactic/semantic categories. The output is passed on to the syntactic analyser which consists of a number spotter, a lexicon and a rule engine. The number spotter spots numbers in the input, indicating, e.g., the user’s age. The lexicon entries consist of syntactic/semantic categories for individual words. After passing through the number spotter and lexicon, the user input is a sequence of semantic and syntactic categories. The rule engine then applies rules defined on the presence of certain semantic/syntactic categories at specific positions in the sequence. The domain/topic spotter spots the input topic(s) by mapping the semantic/syntactic categories to their respective topics. The mapping is defined at design time. Domains are identified based on topics. The result is sent to the FSA (Finite State Automaton) processor which acts as the deepest level of parsing. If the user sequence is able to traverse an FSA, the result corresponding to that FSA is the NLU output semantics. The FSAs are developed off-line from a training corpus. The result consisting of domain(s), topic(s) and semantics is sent to the input fusion module which forwards the result to the character module.

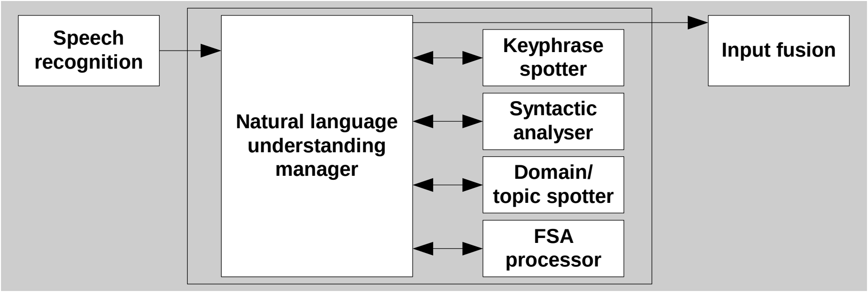

Figure 2. NICE HCA natural language understanding module.

The HCA character module (Figure 3) is managed by the character module manager which also takes care of module-external communication. Viewed as a whole, the character module is in one of three output states, producing either: non-communicative action output, communicative function output, or communicative action. Non-communicative action (NCA) output is produced when nobody is talking to HCA. In this state, he is simply doing his work in his study. Communicative function (CF) output is produced when someone is talking and/or gesturing to HCA, to which he responds by showing awareness of the user's input. For this to happen in real time, the character module has fast-track connections to the speech and gesture recognisers in order to act as soon as one of them receives input. Communicative action (CA) output is HCA's conversational contributions. The overall state relationships are: NCA -> CF <-> CA -> NCA. Thus, NCA, the system's "resting state", must be followed by a CF state in which a new user starts addressing HCA. Following the user's first conversational contribution, conversation in which user and system take turns (CF <-> CA) may go on for a while, eventually being followed by the NCA state.

The mind-state agent (MSA, Figure 3) manages the user's spoken and/or gesture input including the planning of which response (or communicative action) to produce to the input. The central module is the MSA Manager (MSAM) which manages the other components of the MSA. Based on proposals from the conversation intention planner which embodies HCA's conversational agenda, the MSAM decides whether to reply to the user's input and/or whether to take the initiative in the conversation. The MSAM contacts the relevant domain agents (DAs) to get a reply and/or a dialogue continuation. For replies, the knowledge base (KB) is always contacted directly. For continuations, the KB is contacted directly unless the proposed output is a mini-dialogue, i.e. a predefined small dialogue, in which case the mini-dialogue processor (MDP) is contacted first. The MDP processes mini-dialogues in a finite state-machine approach. The KB is a database which maintains the system's ontology including references to all of HCA's coordinated spoken and non-verbal output. The retrieved output references are sent to response generation via the MSAM and the character module manager.

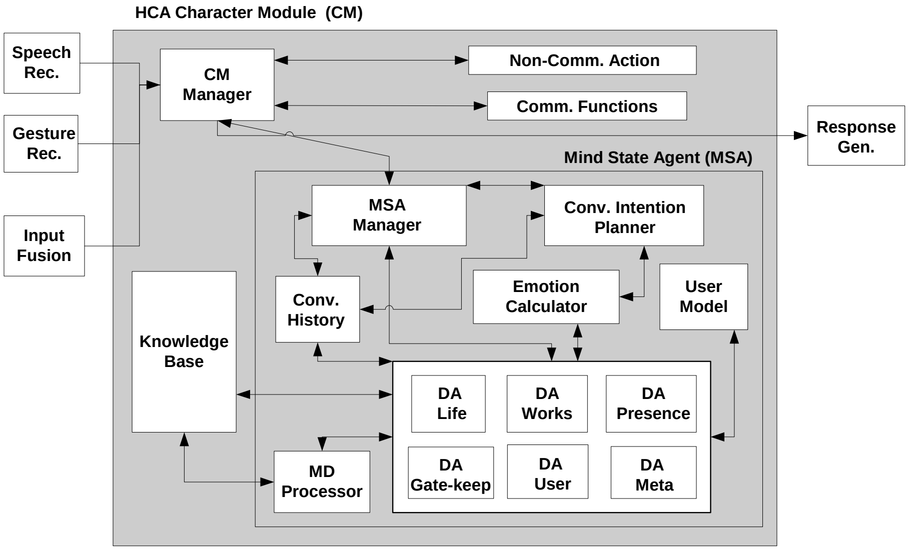

Figure 3. HCA character module architecture.

The emotion calculator updates HCA's emotional state whenever the user's input produces an emotion increment which makes HCA more happy, sad, or angry. The user model stores the information which HCA collects about the present user, i.e. age, gender and nationality, for use during conversation. The conversation history includes a comprehensive record of the conversation per input and output turn.

The response generator receives a parameterised semantic instruction composed of input values, text-to-speech references and/or references to non-verbal behaviours. The TTS references are used to retrieve text template output with embedded start and end tags for non-verbal behaviours (bookmarks). Input values are inserted into the templates, creating a surface language string. The result is sent to the speech synthesiser which synthesises the verbal output and, whenever it meets a bookmark, sends a message to the response generator that now the corresponding non-verbal output descriptions must be sent to the animation module which takes care of the graphics output. The first NICE HCA prototype uses approx. 300 spoken utterance types and 100 different non-verbal behaviour primitives.

The promising results from the January 2004 user tests will be reported elsewhere.

Acknowledgement and references

The work presented is being supported by the EU Human Language Technologies programme under contract IST-2001-35293. We gratefully acknowledge the support.

1. Bernsen, N.O., Dybkjær, H. and Dybkjær, L.: Designing Interactive Speech Systems. From First Ideas to User Testing. Springer Verlag 1998.
2. Cassell, J., Sullivan, J., Prevost, S., and Churchill, E. (Eds.): Embodied conversational agents. Cambridge, MS: MIT Press 2000.