2. History of Conversational System Development

Toyoaki Nishida
Kyoto University
Embodied Conversational Agents (ECAs)

“computer interfaces that can hold up their end of the conversation, interfaces that realize conversational behaviors as a function of the demands of dialogue and also as a function of emotion, personality, and social conversation.” [Cassell, Sullivan, et al., 2000]
History of conversational systems development

- Natural language dialogue systems
- Natural language question answering systems
- Story Understanding systems
- Speech dialogue systems
- Multi-modal dialogue systems
- Embodied Conversational Agents / Intelligent Virtual Human
- The Knowledge Navigator
- Interactional systems
- Conversational Systems
- Transactional systems
- Cognitive systems
- Affective Computing

Timeline:
- 1970
- 1980
- 1990
- 2000
- 2010
Early Natural Language Dialogue Systems

**Baseball**

- One of the earliest natural language question answering system.
- Answers such questions about baseball games.

**Specification list:**

- “Where did the Red Sox play on July 7?”
  - Place = ?
  - Team = Red Sox
  - Month = July
  - Day = 7

- “What teams won 10 games in July?”
  - Team(winning) = ?
  - Game(number of) = 10
  - Month = July

**Dictionary:**

- “team” -> meaning: Team=(blank)
- “Red Sox” -> meaning: Team=Red Sox
- “who” -> meaning: Team=?
- “winning” -> meaning: subroutine

**Data:**

- Month = July
- Place = Boston
- Day = 7
- Game Serial No. = 96
- (Team = Red Sox, Score = 5)
- (Team = Yankees, Score = 3)

[Green 1961]
Early Natural Language Dialogue Systems

**Modules of Baseball**

1. Question Read-in
2. Dictionary Look-up
3. Syntax
   (1) scan for ambiguities in part of speech: in some cases, resolved by looking at adjoining words; in other cases, resolved by inspecting the entire question.
   (2) locates and brackets the noun phrases, [], and the prepositional and adverbial phrases, (). The verb is left unbracketed. E.g.,
   “How many games did the Yankees play in July?”
   -> [How many games] did [the Yankees] play (in [July])?
   
   Any unbracketed preposition is attached to the first noun phrase in the sentence, and prepositional brackets added. E.g.,
   “Who did the Red Sox lose to on July 5?”
   -> (To [who]) did [the Red Sox] lose (on [July 5])?

The syntax routine determines whether the verb is active or passive and locates its subject and object.

The syntax analysis checks to see if any of the words is marked as a question word. If not, a signal is set to indicate that the question requires a yes/no answer

4. Content Analysis
   The content analysis uses the dictionary meanings and the results of the syntactic analysis to set up a specification list for the processing program. E.g.,
   “each team”: Team=(blank) -> Team=each
   “what team”: Team=(blank) -> Team=
   “Who beat the Yankees on July 4”: Team=(blank) -> Team=?, Team(winning)=? Team(losing)=Yankees
   “six games”: Game=(blank)->Game(number of)=6
   “how many games”: Game=(blank)->Game(number of)=?
   “Who was the winning team...”: Team=? and Team(winning)=(blank)
   -> Team(winning)=?

5. Processing
   The specification list indicates to the processor what part of the stored data is relevant for answering the input question. The processor extracts the matching information from the data and procedures, for the responder, the answer to the question in the form of a list structure.

[Weizenbaum 1966]
Early Natural Language Dialogue Systems

LUNAR

Prototype natural language question answering system that helps lunar geologists access chemical analysis data on lunar rock and soil composition.

(i) Syntactic analysis using heuristic/semantic information to choose the most likely parsing (Augmented Transition Network Grammar was used)
(ii) Semantic interpretation: produce a formal representation for queries
(iii) Execution of this formal expression in the retrieval

[Woods 1973]
Early Natural Language Dialogue Systems

Augmented Transition Network Grammar

S:

NP → V → NP
  ↓    
  AUX  V
  ↑    

NP:

DET → ADJ
  ↓    
  NOUN
  ↓    
  PRON

PP:

PREP → NP
Early Natural Language Dialogue Systems

**SHRDLU**

- Natural language understanding system working on the “Blocks” world.
- Based on the belief that “a computer cannot deal with language unless it can understand the subject it is discussing”
- Answers questions, executes commands, and accepts information in normal English dialog.
- Uses semantic information and context to understand discourse and disambiguate sentences.
- Procedural knowledge representation <-> declarative KR

Early Natural Language Dialogue Systems

USER: “Pick up a big red block”
SHRDLU: “OK”

Early Natural Language Dialogue Systems

**ELIZA**: Natural language dialogue system without explicit knowledge about the discourse domain

Transformation rules: collection of key lists.

\[
(K \ ((D_1) \ (R_{1,1}) \ (R_{1,2}) \cdots (R_{1,m_1}))
\]

\[
((D_2) \ (R_{2,1}) \ (R_{2,2}) \cdots (R_{2,m_2}))
\]

\[
\cdots
\]

\[
((D_n) \ (R_{n,1}) \ (R_{n,2}) \cdots (R_{n,m_n}))
\]

where, \( K \): keyword, \( D_i \): decomposition template, \( R_{i,j} \): Reassembly rule.

Example of a decomposition template and a reassembly rule:

D: (0 YOU 0 ME)

R: (WHAT MAKES YOU THINK I 3 YOU)

“It seems that you hate me”

-> “What makes you think I hate you”

Substitution rule. E.g.,

(YOURSELF = MYSELF)

(MY YOUR 5 (transformation rules))
The Hearsay-II Speech-Understanding System

Integrates multiple levels of information processing by knowledge sources coordinated by the blackboard model:

- Parameter
- Segment
- Syllable
- Word
- Word-sequence
- Phrase
- Data base interface

Combining top-down (hypothesis driven) and bottom-up (data-driven) processing.

Selective attention to allocate limited computing resources.

[Erman 1980]
The blackboard architecture

(a) Parameters

(b) Segments

(c) Syllable classes

(d) Words – 2

(e) Words - 1

(f) Word sequences

(g) Phrases

[Erman 1980]
The blackboard architecture

- Blackboard
- Level_1
- Level_2
- Level_k
- KS_1
- KS_n
- Blackboard Monitor
- Scheduling queues
- Focus-of-control Database
- Scheduler

Program modules
Databases
Data flow
Control flow

[Erman 1980]
PUT-THERE-THERE

The conjoint use of voice-input and gesture recognition to command events on a large format raster-scan graphic display in a “Media Room”

Commands

“Create a blue square there.”
“Move the blue triangle to the right of the green square”
“Move that to the right of the green square.” (with pointing)
“Put that there” (indicated by gesture)
“Make that smaller” (with pointing gesture)
“Make that (indicating some item) like that (indicating some other item)”
“Delete that” (pointing to some item)
“Call that ... the calendar” (with pointing)
“Phil”: concrete image of a personal assistant.
Social responses to computers

- not: conscious beliefs that computers are human or human-like.
- not: from user’s ignorance or psychological or social dysfunctions
- not from: a belief that subjects are interacting with programmers
- rather: human-computer relationship is fundamentally social.

Nass 1994]
Requirements

- **Support interactive give and take**
  Assistants will need not only respond to questions but also ask questions

- **Recognize the costs of interaction and delay**
  It is inappropriate to require the user’s confirmation of every decision made while carrying out a task.

- **Manage interruptions effectively**
  When it is necessary to initiate an interaction with the user, the assistant needs to do so carefully, recognizing the likelihood that the user is already occupied to some degree.

- **Acknowledge the social and emotional aspects of interaction**
  To become a comfortable working partner, a computer assistant will need to vary its behavior depending on such the task, the time of day, and the boss’s mood.

=> The PERSONA project @ Microsoft Research

[Ball 1997]
The architecture of Peedy

Spoken Language Interpretation

Character Animation

Background and Object Animation

Sound Generation

Dialogue Management

[Ball 1997]
The architecture of Peedy

Microphone
  whisper
  speech
  recognition

Names
  proper name
  substitution

NLP
  language analysis

Semantic
  template matching
  object descriptions

Names database

Action templates database

Dialogue
  context & conversation state
  rules database

Player/ReActor
  animation engine

Speech & animation database

Speech controller

Sound

CD player

Character

Object database (CDs)

Application
  CD changes

[Ball 1997]
A state transition machine model with 5 conversational states and 17 input events is used.

Peedy: “I have The Bonnie Raitt Collection, would you like to hear something from that?”

User: “Sure”

pePickTrack; genTracks; doSelect; Say <!PDSays(how About)>
Believability

Believable agents

provides the illusion of life, and permits the audience’s suspension of disbelief.
... critical in theater, film, animation, radio drama, ...

Questions

i. What makes characters believable? What makes agents believable?
   Are agents different from characters?

ii. What is the nature of current integrated architectures and situated agents?
    How well do they support believability? Are there clear areas demanding study?

iii. How much breadth of capability is necessary to produce a believable agent?
     How much depth (competence) is necessary?

iv. What are the respective role of movement and language in achieving believability?

v. What is the role of context in establishing expectations in the user and thus is simplifying the
   task? For instance, how can the setting of an interface agent or the theme of a story help
   limit the technical requirements on agents?

vi. How can we measure believability and progress toward believability?
    Why don’t artists use “scientifically valid” techniques to evaluate the believability of their
    characters?

[Bates 1994]
Interactive Drama:
Interactive drama concerns itself with building dramatically interesting virtual worlds inhabited by computer-controlled characters, within which the user (the player) experiences a story from a first person perspective. ... (Bates 1992)

Drama = Character + Story + Presentation
Personality
Rich personality should infuse everything that a character does. What makes characters interesting are their unique ways doing things.

Emotion
Characters exhibit their own emotions and respond to the emotions of others in personality-specific ways.

Self-motivation
Characters have their own internal drives and desires which they pursue whether or not others are interacting with them.

Change
Characters grow and change with time, in a manner consistent with their personality.

Social relationships
Characters engage in interactions with others in a manner consistent with their relationship. In turn, these relationships change as a result of the interaction.

Illusion of life
Pursuing multiple, simultaneous goals and actions, having broad capabilities, and reacting quickly to stimuli in the environment.

[Mateas 1997]
Implements the social, linguistic, and psychological conventions of conversation.

- Has a human-like body. Uses eye gaze, body posture, hand gestures, and facial displays to organize and regulate the conversation.
- The conversational model relies on the function of non-verbal behaviors as well as speech.
- A full symmetry between input and output modalities: not only respond to visual, audio and speech cues (such as speech, shifts in gaze, gesture, and non-speech audio but also generate these cues.

[Cassell 1999]
Rea

[Cassell 1999]
BEAT (Behavior Expression Animation Toolkit)

Discourse Model

Knowledge Base

Word timing

Behavior Generation

Behavior Suggestion

Behavior Selection

Generator Set

Filter Set

Translator

Text Input

Animation

[ Cassell 2004 ]
"Mary socked John."
"Mary punched John."
"Mary hit John with her fist."
"John was socked by Mary."
"Marie a donne un coup de poing a Jean."
"Maria pego a Juan."

**Margie**
Action: PROPEL
Actor: Mary
Object: Fist
From: Mary
To: John

<table>
<thead>
<tr>
<th>Primitive</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATRANS</td>
<td>Transfer of an abstract relationship (i.e. give)</td>
</tr>
<tr>
<td>PTRANS</td>
<td>Transfer of the physical location of an object (e.g., go)</td>
</tr>
<tr>
<td>PROPEL</td>
<td>Application of physical force to an object (e.g. push)</td>
</tr>
<tr>
<td>MOVE</td>
<td>Movement of a body part by its owner (e.g. kick)</td>
</tr>
<tr>
<td>GRASP</td>
<td>Grasping of an object by an action (e.g. throw)</td>
</tr>
<tr>
<td>INGEST</td>
<td>Ingesting of an object by an animal (e.g. eat)</td>
</tr>
<tr>
<td>EXPEL</td>
<td>Expulsion of something from the body of an animal (e.g. cry)</td>
</tr>
<tr>
<td>MTRANS</td>
<td>Transfer of mental information (e.g. tell)</td>
</tr>
<tr>
<td>MBUILD</td>
<td>Building new information out of old (e.g. decide)</td>
</tr>
<tr>
<td>SPEAK</td>
<td>Producing of sounds (e.g. say)</td>
</tr>
<tr>
<td>ATTEND</td>
<td>Focusing of a sense organ toward a stimulus (e.g. listen)</td>
</tr>
</tbody>
</table>

[Schank 1975]
Script-based understanding

SAM (Script Applier Mechanism)

FRUMP
Story understanding and generation

Plan-based understanding

PAM

POLITICS

Dynamic Memory

IPP

BORIS

CYRUS

Story telling

TALE-SPIN
1. Conversational artifacts have a long history of development.
2. Conversational systems evolved from natural language dialogue systems. Multiple modality and embodiment have been incorporated, then.
3. The Knowledge Navigator gave a significant impact on the concept of intelligent virtual agents.
4. Personality, social interactions and lifelikeness are key features of intelligent virtual agents.
5. In order to create natural conversational behaviors, conversational artifacts need to coordinate verbal and nonverbal behaviors.
6. Story understanding and generation techniques are necessary for processing content.
References


