3. Methodologies for Conversational System Development

Toyoaki Nishida
Kyoto University
FearNot! (Fun with Empathic Agents to Reach Novel Outcomes in Teaching) is a Virtual Learning Environment (VLE) for helping school children learn a bullying scenario, by allowing them to explore what happens in bullying [Aylett 2005 IVA]. FearNot! integrates an appraisal-based emotion engine and a coping mechanism for handling the given problem. Emotions not only influence the agents’ reactive behavior, but also guide the planning process. The child user watches the interaction and is asked to give an advice for the victim about how to deal with the situation [Aylett 2012].
The space of issues

Research

Development

Conversation environment

Autonomous conversational agent

- Internal representations
- Knowledge and models
  - affective computing / theory of mind
  - dialogue management
  - interpretation
  - generation
  - sound processing
  - computer vision/graphics
  - signal processing
  - speech
  - visual
  - haptic and other modalities

Users or their avatars

modeling and content management

design and evaluation
Technical basis for conversational informatics

- Architecture
- Scripts and Markup Languages
- Corpus-based approach
- Evaluation

[Nishida 2014]
Story Understanding and Generation

- Episodic memory
- Semantic memory
- Story understander
- Story generator
- Learner
- Dynamic memory

Goal: stories [Nishida 2014]
Dialogue Management

Task

Intended meaning

Discourse analysis

Syntax and semantic analysis

Discourse planning

Sentence generation

Discourse

Utterances

[Nishida 2014]
BDI-architecture

incoming signals

Sensors

Beliefs

Reasoner

Desires

Intentions

Plans

outgoing effects

[Georgeff 1990]
Affective Computing

Cognitive processing

High level
Inference and decision making

Low level
Pattern recognition and synthesis

Representations / signals

Representations / signals

Emotional states

[Picard 1997]
Cognitive architecture

Emotion module
- dominance
- mapping
- arousal
- PAD space
- mood
- awareness likelihood

Cognition module
- conscious appraisal
- Reasoning layer
  - social context
  - reappraisal (coping)
- secondary emotions
- aware emotions
- non-conscious appraisal
- primary emotions
- Reactive layer
  - elicitation
- emotions X awareness

Perception
- valence
- dominance

Action

[Becker-Asano 2008]
Cognitive architecture

Reactive layer

Goals

Intentions

Focus and Coping

Intention generation

Intention selection

Personality thresholds

Deliberative layer

Autobiographic memory

Energy Integrity Affiliation Certainty Competence

Modulating parameters

PSI

Appraisal

Effectors

Action

Sensors

Perception

Emotional state

Thresholds

Properties

Relations

Knowledge base

Appraisal

Action tendencies

[Lim 2010]
Cognitive architecture

Deliberative layer

- Goals
  - Intention generation
  - Intention selection
  - Focus and Coping

Mental model

- Appraisal

Reactive layer

- Interpretation
  - Appraisal
  - Action tendencies

- Sensors
  - Effectors

Realworld

Memory

Knowledge

[Nishida 2014]
Requirements at the platform level

- The system should be able to run multiple processes simultaneously to handle multimodal signals.
- The system should allow the programmer to write a complex control structure, without sacrificing real-time response.
- The system should allow the programmer to write codes across different levels of abstraction, ranging from signals to semantic representation.
- The system should allow the programmer to use high level languages to write her/his idea in multiple levels of abstraction.
- The system architecture should allow the system to be easily scale-up.

[Nishida 2014]
Hierarchical processing

- Symbolic Representation
  - Parametric Representation
    - Input
    - Output
  - Signal
- Parametric representation from sensors
- Parametric representation for motors
Authoring Embodied Conversational Agents

Script languages / markup languages for building Believable synthetic characters

- Synchronizing speech, eye gaze, gesture
- Representing personality or emotional states by gestural behaviors, facial expression, utterances, etc.
- Coordinating body motions of multiple characters (in conversations)
- Communication between users and other agents

[Prendinger 2004]
Script vs Mark-up Language

Script

Interpreter

animation

Action Planner

Knowledge base

Constraints

Action Realizer

animation

Mark-up Language

[Nishida 2014]
Scripts

Specifying behaviors of ECA’s, communicative acts like gestures and postures, in particular.

Principles

- **Convenience**
  Hides geometric difficulties so that even the authors who have limited knowledge of computer graphics can use it in a natural way.

- **Compositional Semantics**
  Specification of composite actions based on existing components.

- **Redefinability**
  Allows the authors to explicitly script actions in terms of other actions.

- **Parameterization**
  Allows the authors to specify actions in terms of how they cause changes over time to each individual degree of freedom.

- **Interaction**
  Scripting actions should be able to interact with the world, including objects and other agents.

[Huang 2004]
AIML (Artificial Intelligence Mark-up Language)

AIML, developed by the Alicebot free software community during 1995-2000, enables people to input knowledge into chat-bots. The basic minimalist approach comes from ELIZA.

Principles

1. Symbolic Reduction: Reduce complex grammatical forms to simpler ones.
2. Divide and Conquer: Split an input into two or more subparts, and combine the responses to each.
3. Synonyms: Map different ways of saying the same thing to the same reply.
4. Spelling or grammar corrections.
5. Detecting keywords anywhere in the input.
6. Conditionals: Certain forms of branching may be implemented with <srai>.
7. Any combination of 1-6.

<srai>: stimulus-response, AI

[Wallace 2003]
C: Knock knock.
R: Who’s there?
C: Banana.
R: Banana who?
C: Knock knock.
R: Who’s there?
C: Banana.
R: Banana who?
C: Knock knock.
R: Who’s there?
C: Orange.
R: Orange who?
C: Orange you glad I didn’t say banana.
R: Ha ha very funny, Nancy.

In AIML the syntax `<that>...</that>` encloses a pattern that matches the robot’s previous utterance.

[Wallace 2003]
Microsoft Agent

Set of programmable software services that supports the presentation of interactive animated characters.

Features

- **Conversational interface**
  leverages natural aspects of human social communication. To be blended with the conventional interface components such as windows, menus, etc.
  Optional support for speech recognition to realize voice commands.
  Characters can respond with synthetic speech, recorded audio or texts.

- **Animated characters**
  appear in their own window. embedded in software with Visual Basic / in web page with VBScript

- **Social Interaction**
  Based on Media Equation. Use nonverbal social cues that convey attitudes, identities, and emotions. Personality, etiquette, praise, team, gender effects in user interface.

[Clark 1998]
Set Genie = Agent1.Characters("Genie")
Set Robby = Agent1.Characters("Robby")
Genie.Show
Genie.Play “Greet”
Genie.Speak "Hello!"
Robby.Show
Robby.MoveTo 100,200
Robby.Play "GestureRight"
Robby.Hide
Genie.Hide

...
MPML

XML-based markup language for scripting character-based multi-modal presentations. Provides appropriate tagging structure that enables authors to specify features of presentations by human presenters.

Principles

- Easy to use
  Not assume programming skills.
- Intelligibility
  Provides names and abbreviations that clearly indicate their meaning.
- Believability
  Facilitates scripting consistent emergent behavior. Making the agent move and speak according to its intended personality.
- Extensibility
  The features not supported by MPML can be added at the cost of scripting in the JavaScript or the Java programming language.
- Easy Distribution
  A converter transforms the MPML script to JavaScript so that it can run on a browser that has JavaScript.

[Prendinger 2004]
Specifying behaviors of ECA’s, communicative acts like gestures and postures, in particular.

Principles

- **Convenience**
  Hides geometric difficulties so that even the authors who have limited knowledge of computer graphics can use it in a natural way.

- **Compositional Semantics**
  Specification of composite actions based on existing components.

- **Redefinability**
  Allows the authors to explicitly script actions in terms of other actions.

- **Parameterization**
  Allows the authors to specify actions in terms of how they cause changes over time to each individual degree of freedom.

- **Interaction**
  Scripting actions should be able to interact with the world, including objects and other agents.

[Huang 2004]
Parameterization

(a) Direction reference for humanoid

(b) Combination of the directions for left arm

(c) Typical joints for humanoid

(d) Elbow joint in different situations

[Huang 2004]
XML-based, embodied agent, character attribute, definition and animation scripting language designed to aid in the rapid incorporation of life-like agents into online applications or virtual reality world.

- **Description**
  - **Classification of motion**
    - Movement, Pointing, Grasping, Gaze, Gesture.
  - **Head gestures taxonomy**
    - Symbolic gesture, Iconic gesture, Deictic gesture.
  - **Hand gestures taxonomy**
    - Posture, Motion, Orientation, Gestlets (A set of high-level tags that are constituted from the lower level tags, to make up gestures like Point, Wave, etc), Fingers.
  - **Body gestures taxonomy**
    - Natural, Relax, Tense, Iconic, Incline.
  - **Emotions**
    - Class, Valence (positive/neutral/negative), Subject, Target, Intensity, Time-stamp, Origin.

[Arafa 2004]
SAIBA framework for multimodal generation

Intent Planning \[\rightarrow\] FML \[\rightarrow\] Behavior Planning \[\rightarrow\] BML \[\rightarrow\] Behavior Realization

FML: Function Markup Language [Heylen 2008]

BML: Behavior Markup Language [Kopp 2006]
The Goals

- A powerful, unifying model of representations for multimodal generation, based on BEAT, MURML, APML, and RRL.

- Application independent, graphics model independent, and to present a clear-cut separation between information types (functions vs behaviors).

- Work together for better multi-modal behaviors

[Kopp 2002]
### BML: Behavior Markup Language

<table>
<thead>
<tr>
<th>g-units</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>g-phrases</td>
<td>Calm</td>
</tr>
<tr>
<td>g-phases</td>
<td>prep</td>
</tr>
</tbody>
</table>

[Kopp 2002]
BML: Behavior Markup Language

Synchronization

The seven synch points

Start, Ready, Stroke-start, Stroke, Stroke-end, Relax, End

Pre-stroke-hold, Post-stroke-hold

[Kopp 2006]
**The BML behavior elements**

<table>
<thead>
<tr>
<th>BML Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;head&gt;</td>
<td>Movement of the head independent of eyes. Type include nodding, shaking, tossing and orienting to a given angle.</td>
</tr>
<tr>
<td>&lt;torso&gt;</td>
<td>Movement of the orientation and shape of the spine and shoulder.</td>
</tr>
<tr>
<td>&lt;face&gt;</td>
<td>Movement of facial muscles to form certain expressions. Types include eyebrow, eyelid and larger expressive mouth movements.</td>
</tr>
<tr>
<td>&lt;gaze&gt;</td>
<td>Coordinated movement of the eyes, neck and head direction, indicating where the character is looking.</td>
</tr>
<tr>
<td>&lt;body&gt;</td>
<td>Full body movement, generally independent of the other behaviors. Types include overall orientation, position and posture.</td>
</tr>
<tr>
<td>&lt;legs&gt;</td>
<td>Movements of the body elements downward from the hip: pelvis, hip, legs including knee, toes and ankle.</td>
</tr>
<tr>
<td>&lt;gesture&gt;</td>
<td>Coordinated movement with arms and hands, including pointing, reaching, emphasizing (beating), depicting and signaling.</td>
</tr>
<tr>
<td>&lt;speech&gt;</td>
<td>Verbal and paraverbal behavior, including the words to be spoken (for example by a speech synthesizer), prosody information and special paralinguistic behaviors (for example filled pauses).</td>
</tr>
<tr>
<td>&lt;lips&gt;</td>
<td>This element is used for controlling lip shapes including the visualization of phonemes.</td>
</tr>
</tbody>
</table>
<wait> can align a behavior with a condition or an event

Example 1

```
<bm1>
  <gesture id="g1" type="point" target="object1"/>
  <body id="b1" posture="sit"/>
  <wait id="w1" condition="g1:end AND b1:end"/>
  <gaze target="object2" start="w1:end"/>
</bm1>
```
<wait> can align a behavior with a condition or an event

Example 2

```
<bml>
  <speech id="s1" type="text/plain">
    First sentence
  </speech>
  <event start="s1:end" emit="ACT1_COMPLETE" />
</bml>

<bml>
  <wait id="w1" event="ACT_COMPLETE" duration="5.0"/>
  <speech type="text/plain" start="w1:end">
    Second sentence
  </speech>
</bml>
```

s1: speech: First sentence

w1: wait

s2: speech: Second sentence

Waits for the “ACT1_COMPLETE” event for maximally 5.0 seconds and speaks the second sentence.

[Kopp 2006]
Timing holds

Example 3

<speech id="s1"> <sync id="1">This or </sync> <sync id="2">that.</sync></speech>

<gesture id="d2" type="DEICTIC" ... stroke="s1:2" />

<gesture id="d1" type="DEICTIC" ... stroke="s1:1" relax="d2:ready" />

[Kopp 2006]
Generating natural gestures

Offline phase workflow

Video corpus → Manual Annotation - speech - gesture → Annotation files → Modeling → Gesture profiles

Gesture lexicon

Animation lexicon

[Kipp 2007]
**Coding scheme**

NOVACO, a coding scheme for speech and gesture, specially designed for the purpose of gesture generation.

<table>
<thead>
<tr>
<th>Speech transcription:</th>
<th>Gesture transcription:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Words and segments</td>
<td>1. Gesture structure</td>
</tr>
<tr>
<td>2. Parts of speech</td>
<td>a. Movement phases</td>
</tr>
<tr>
<td>3. Theme/Rheme and focus</td>
<td>b. Movement phrases</td>
</tr>
<tr>
<td>4. Discourse relations</td>
<td>2. Gesture classes</td>
</tr>
<tr>
<td></td>
<td>1. Adaptors</td>
</tr>
<tr>
<td></td>
<td>2. Emblems</td>
</tr>
<tr>
<td></td>
<td>3. Deictics</td>
</tr>
<tr>
<td></td>
<td>4. Iconics</td>
</tr>
<tr>
<td></td>
<td>5. Metaphorics</td>
</tr>
<tr>
<td></td>
<td>6. Beats</td>
</tr>
<tr>
<td></td>
<td>3. Gesture lexicon and lemmas</td>
</tr>
<tr>
<td></td>
<td>4. Gesture properties</td>
</tr>
</tbody>
</table>

[Ch7-8, Kipp 2003]
(1) Capturing temporal structure

<table>
<thead>
<tr>
<th>g-units</th>
<th>Calm</th>
<th>Cup</th>
<th>Wipe</th>
</tr>
</thead>
<tbody>
<tr>
<td>g-phrases</td>
<td>prep</td>
<td>stroke</td>
<td>hold</td>
</tr>
</tbody>
</table>

[Kipp 2007 JLRE]
(2) Capturing spatial form

Height: \{Above head, Head, Shoulder, Chest, Abdomen, Belt, Below belt\},
Distance: \{Far, Normal, Close, Touch\},
Radial orientation: \{Inward, front side, out, far out\},
Arm swivel: \{Touch, Normal, Out, Orthogonal\}
Gesture notation conventions

Handedness ∈ \{BH, LH, RH\}

HandOrientation ∈ \{P, F\} × \{TC, AC, UP, DN, TB, AB\}

HandPosition ∈ \{CC\} ⊔ \{C, P, EP\} × \{UP, UR, UL, RT, LT, LR, LL, LW\}

[McNeill 1992 Hand and gesture]
(3) Capturing membership to lexical category

The imaginary “circle” may be traced multiple times while potentially moving and/or becoming bigger/smaller in the process. If performed with two hands, the hands can move in a symmetrical fashion or alternate (i.e. the hands revolve around each other) like in this sample.”

[Kipp 2007 JLRE]
Generating natural gestures

Online processing

Input text

- Input Preproc.
- Gesture creation
- G-unit creation
- G-unit planning

Gesture profiles

Generation

Animation lexicon

Animation Engine

Gesture script

Animated character

Offline processing

[Kipp 2007]
Requirements as a video annotation tool

- Integrated video player
- Intuitive visualization
- Multiple layers
- Genericness
- Temporal inclusion relationship between layers
- Efficient coding interface
- XML data format
- Import/export facilities
- Cross-level coding
- Online and offline access to coding scheme
- Management tool for multiple annotations
- Search facilities
- Non-temporal elements
Gesture generation by imitation

Terms and levels of abstraction in annotation

\[\text{tool}\]

File format \(\leftarrow\) Is expressed in Meta-scheme \(\rightarrow\) implements Anvil

provides syntax for provides components for the definition of

coding scheme \(\rightarrow\) must define project

\[\text{user}\]

annotation file \(\leftarrow\) In written into annotation \(\rightarrow\) adds to video coder

\[\text{[Kipp 2003]}\]
## Coding reliability

<table>
<thead>
<tr>
<th></th>
<th>$c_1$</th>
<th>$c_2$</th>
<th>\ldots</th>
<th>$c_n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c_1$</td>
<td>$f_{1,1}$</td>
<td>$f_{1,2}$</td>
<td>\ldots</td>
<td>$f_{1,n}$</td>
</tr>
<tr>
<td>$c_2$</td>
<td>$f_{2,1}$</td>
<td>$f_{2,2}$</td>
<td>\ldots</td>
<td>$f_{2,n}$</td>
</tr>
<tr>
<td>\vdots</td>
<td>\vdots</td>
<td>\vdots</td>
<td>\ddots</td>
<td>\vdots</td>
</tr>
<tr>
<td>$c_n$</td>
<td>$f_{n,1}$</td>
<td>$f_{n,2}$</td>
<td>\ldots</td>
<td>$f_{n,n}$</td>
</tr>
<tr>
<td></td>
<td>$f_{1}$</td>
<td>$f_{2}$</td>
<td>\ldots</td>
<td>$f_{n}$</td>
</tr>
</tbody>
</table>

### Segmentation reliability

\[
\text{hits} \quad \frac{\text{hists} + \text{misses}}
\]

### Classification reliability

$\kappa$

\[ p_0 = \frac{\sum_{i=1}^{n} f_{i,i}}{N} \]
\[ p_e = \frac{\sum_{i=1}^{n} f_{i,.} f_{.,i}}{N^2} \]
\[ \kappa = \frac{p_0 - p_e}{1 - p_e} \]

[p. 160, Kipp 2003]
Adequacy of a Model

(1) Holdout (test sample estimation)

\[ acc_H = \frac{1}{h} \sum_{i \in D_h} \delta(\mathcal{I}(D_t, y_i), y_i) \]

(2) k-fold Cross Validation

\[ acc_{CV} = \frac{1}{n} \sum_{i \in D} \delta(\mathcal{I}(D_t - D(i), y_i), y_i) \]

[ Kohavi 1995 ]
Direction giving by speech and gesture

How can iconic gestures express meaning? -- how images are mentally linked to referents and how gestures can depict images.

IDF (Image Description Feature)

- Gesture morphology
- IDF: corresponding salient visual characteristic
- Concrete referent

Hand orientation defined in terms of extended finger direction and palm direction

A qualitative, feature-based framework for describing morphological features encoding meaningful geometric and spatial properties.

[Kopp 2007]
Direction giving by speech and gesture

Coverbal gesture on “There’s a church”
In conversation, the listeners’ attention to a shared reference serves as positive feedback to the speaker, and attentional behaviors are indispensable as communicative signals in face-to-face conversation.

(A) Attention towards the physical world:
   (A1) From an ECA to a physical object
   (A2) From a user to a physical object
(B) Attention towards the virtual world:
   (B1) From an ECA to a virtual object
   (B2) From a user to a virtual object

Because ECAs and users inhabit different worlds, sharing a reference between them is difficult. To resolve this problem, it is indispensable to seamlessly integrate these two different worlds and broaden the communication environment in human–agent interaction.

[Nakano 2007]
Data Coding
As a unit of verbal behavior, we tokenized a turn into utterance units (UU) (Nakatani and Traum 1999), corresponding to a single intonational phrase (Pierrehumbert 1980). Each UU was categorized using the DAMSL coding scheme (Allen and Core 1997). In the statistical analysis, we concentrated on the following four categories with regular occurrence in our data: Acknowledgment, Answer, Information request (Info-req), and Assertion. For nonverbal behaviors, the following four types of nonverbal behaviors were coded:
- Gaze at partner (gP): Looking at the partner’s eyes, eye region, or face
- Gaze at map (gM): Looking at the map
- Gaze elsewhere (gE): Looking away elsewhere
- Head nod (Nod): Head moves up and down in a single continuous movement on a vertical axis, but eyes do not go above the horizontal axis.
Grounding judgment by user’s nonverbal signals

**Step 1: Preparing for the next UU.** By referring to the Agenda, the DM can identify the speech act type of the next UU, which is the key to specifying the positive/negative evidence in grounding judgment later.

**Step 2: Monitoring.** The GrM sends the next UU to GM, and the GM begins to process the UU. At this time, the GM logs the start time in the Discourse Model. When it finishes processing (as it sends the final command to the animation module), it logs the end time. The GrM waits for this speech and animation to end (by polling the Discourse Model until the end time is available), at which point it retrieves the timing data for the UU, in the form of time-stamps for the UU start and finish. This timing data is used in the following Judging step.

**Step 3: Judging.** When the GrM receives an end signal from the GM, it starts the judgment of grounding using nonverbal behaviors logged in the Discourse History. Looking up the Grounding Model by the type of the UU, the GrM identifies the positive/negative evidence in grounding judgment.

[Nakano 2007]
CUBE-G Project

**CULture-adaptive BEnavior Generation (CUBE-G)** for interactions with embodied conversational agents

We investigate whether and how the non-verbal behavior of agents can be generated from a parameterized computational model. Specifying a culture’s position on the basic dimensions allows the system to generate appropriate non-verbal behaviors for the agents. The project combines a top down model-based approach with a bottom-up corpus-based approach which allows empirically grounding the model in the specific behavior of two cultures (Japanese and German), and challenges the following objectives:

1. To investigate how to extract culture-specific behaviors from corpora.
2. To develop an approach to multimodal behavior generation that is able to reflect culture specific aspects.
3. To demonstrate the model in suitable application scenarios.

[Matthias Rehm; Elisabeth André; Yukiko Nakano; Toyoaki Nishida 2009]
Dimensions of evaluating embodied conversational agents

- **Usability**
  - Learnability, memorizability, and ease of use
  - Efficiency
  - Errors

- **User perception**
  - Satisfaction
  - Engagement
  - Helpfulness
  - Naturalness and believability
  - Trust
  - Perceived task difficulty
  - Likeability

[Ruttkay 2004]
Data Collection methods

- Qualitative methods
  - Interview and focus group
  - Informal or descriptive observation

- Quantitative methods
  - Questionnaires
  - Systematic observation
  - Log files
  - Heuristic evaluation
  - Biological measures

[Christoph 2004]
Implicit Association Test

The IAT procedure seeks to measure implicit attitudes by measuring their underlying automatic evaluation. The IAT is therefore similar in intent to cognitive priming procedures for measuring automatic affect or attitude.

“In the IAT a subject responds to a series of items that are to be classified into four categories – typically, two representing a concept discrimination such as flowers versus insects and two representing an attribute discrimination such as pleasant versus unpleasant valence. Subjects are asked to respond rapidly with a right-hand key press to items representing one concept and one attribute (e.g., insects and pleasant), and with a left-hand key press to items from the remaining two categories (e.g., flowers and unpleasant). Subjects then perform a second task in which the key assignments for one of the pairs is switched (such that flowers and pleasant share a response, likewise insects and unpleasant). The IAT produces measures derived from latencies of responses to these two tasks. These measures are interpreted in terms of association strengths by assuming that subjects respond more rapidly when the concept and attribute mapped onto the same response are strongly associated (e.g., flowers and pleasant) than when they are weakly associated (e.g., insects and pleasant).”

[Greenwald 1998]
Implicit Association Test

Screen

Word1 (e.g., Music) or Pleasant

Word2 (e.g., Sport) or Unpleasant

Stimulus

Keyboard

Left

Right

[Greenwald 2003]
### Implicit Association Test

<table>
<thead>
<tr>
<th>block</th>
<th>Number of trials</th>
<th>Practice/Test</th>
<th>Items assigned to left-key response</th>
<th>Items assigned to right-key response</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20</td>
<td>Practice</td>
<td>Music</td>
<td>Sport</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>Practice</td>
<td>Pleasant</td>
<td>Unpleasant</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>Practice</td>
<td>Music+Pleasant</td>
<td>Sport+Unpleasant</td>
</tr>
<tr>
<td>4</td>
<td>40</td>
<td>Test</td>
<td>Music+Pleasant</td>
<td>Sport+Unpleasant</td>
</tr>
<tr>
<td>5</td>
<td>20</td>
<td>Practice</td>
<td>Sport</td>
<td>Music</td>
</tr>
<tr>
<td>6</td>
<td>20</td>
<td>Practice</td>
<td>Pleasant</td>
<td>Unpleasant</td>
</tr>
<tr>
<td>7</td>
<td>40</td>
<td>Test</td>
<td>Sport+Pleasant</td>
<td>Music+Unpleasant</td>
</tr>
</tbody>
</table>

\[
\text{Score} = \frac{1}{2} \left( \frac{\text{AverageLatencyInBlock k6} - \text{AverageLatencyInBlock k3}}{\text{SD for Latency in Block 6 and Block 3}} \right) + \frac{1}{2} \left( \frac{\text{AverageLatencyInBlock k7} - \text{AverageLatencyInBlock k4}}{\text{SD for Latency in Block 7 and Block 4}} \right)
\]

[Greenwald 2003]
1. Standard methodologies for conversational system development today include architecture, script and markup languages, corpus-based system development, and evaluation.

2. At the architecture level, the key issue is to identify the cognitive components and their interconnections.

3. Script language allows one to code the behaviors of embodied conversational agents without worrying about implementation details.

4. Markup languages are generally less procedural. Although some sophisticated mechanism may be needed, they allow for specifying the behaviors just with specifying constraints.

5. Corpus-based approaches are needed to care about subtleties and personalities.

6. Evaluation is necessary for turning hand-crafting practice into science and technology.
References


References


