An assistive handwashing system with emotional intelligence

Luyuan Lin

University of Waterloo

Supervisor:
Jesse Hoey

July 22, 2014

Overview

Problem Statement - Motivation

The COACH system

- is an assistive system helping with an elder's daily activities
- monitors a user washing his/her hands
- detects when the user has lost track of what he/she is doing
- displays a prerecorded assistive prompt when needed
- works well for some persons, but not as well for others

Problem Statement - Motivation

The COACH system

- is an assistive system helping with an elder's daily activities
- monitors a user washing his/her hands
- detects when the user has lost track of what he/she is doing
- displays a prerecorded assistive prompt when needed
- works well for some persons, but not as well for others

Using Emotional Intelligence in Assitive Systems

Problem Statement - Motivation

The COACH system

- is an assistive system helping with an elder's daily activities
- monitors a user washing his/her hands
- detects when the user has lost track of what he/she is doing
- displays a prerecorded assistive prompt when needed
- works well for some persons, but not as well for others

Using Emotional Intelligence in Assitive Systems

- recognization of affective states

Problem Statement - Motivation

The COACH system

- is an assistive system helping with an elder's daily activities
- monitors a user washing his/her hands
- detects when the user has lost track of what he/she is doing
- displays a prerecorded assistive prompt when needed
- works well for some persons, but not as well for others

Using Emotional Intelligence in Assitive Systems

- recognization of affective states
- generation of affective signals

Problem Statement - Motivation

The COACH system

- is an assistive system helping with an elder's daily activities
- monitors a user washing his/her hands
- detects when the user has lost track of what he/she is doing
- displays a prerecorded assistive prompt when needed
- works well for some persons, but not as well for others

Using Emotional Intelligence in Assitive Systems

- recognization of affective states
- generation of affective signals
- study of human emotions

Problem Statement - Motivation

The COACH system

- is an assistive system helping with an elder's daily activities
- monitors a user washing his/her hands
- detects when the user has lost track of what he/she is doing
- displays a prerecorded assistive prompt when needed
- works well for some persons, but not as well for others

Using Emotional Intelligence in Assitive Systems

- recognization of affective states
- generation of affective signals
- study of human emotions
- computationally modelling affective HCIs

Problem Statement - Objectives

To augment the COACH system with an emotional reasoning engine based on BayesACT so that the augmented system:

Problem Statement - Objectives

To augment the COACH system with an emotional reasoning engine based on BayesACT so that the augmented system:

- is designed in a portable and extensible way

Problem Statement - Objectives

To augment the COACH system with an emotional reasoning engine based on BayesACT so that the augmented system:

- is designed in a portable and extensible way
- runs in real-time from the perspective of the user group

Problem Statement - Objectives

To augment the COACH system with an emotional reasoning engine based on BayesACT so that the augmented system:

- is designed in a portable and extensible way
- runs in real-time from the perspective of the user group
- provides at least a level of functional assistance of as high quality as the COACH

Problem Statement - Objectives

To augment the COACH system with an emotional reasoning engine based on BayesACT so that the augmented system:

- is designed in a portable and extensible way
- runs in real-time from the perspective of the user group
- provides at least a level of functional assistance of as high quality as the COACH
- is able to tune the prompts in some way according to the emotional state of a user

Problem Statement - Objectives

To augment the COACH system with an emotional reasoning engine based on BayesACT so that the augmented system:

- is designed in a portable and extensible way
- runs in real-time from the perspective of the user group
- provides at least a level of functional assistance of as high quality as the COACH
- is able to tune the prompts in some way according to the emotional state of a user

Note: The last objective is ill-defined, as the question of how exactly tuning prompts to users will be most effective is not clear at this point.

Concepts - ACT

Affect Control Theory (ACT)

- represents emotions as vectors that represent evaluation (E), potency (P), and activity (A) respectively

Concepts - ACT

Affect Control Theory (ACT)

- represents emotions as vectors that represent evaluation (E), potency (P), and activity (A) respectively
- describes social events by an Actor-Behaviour-Object (ABO) grammar

Concepts - ACT

Affect Control Theory (ACT)

- represents emotions as vectors that represent evaluation (E), potency (P), and activity (A) respectively
- describes social events by an Actor-Behaviour-Object (ABO) grammar
- "fundamentals" of identities and behaviours; shared between people within a same culture

Concepts - ACT

Affect Control Theory (ACT)

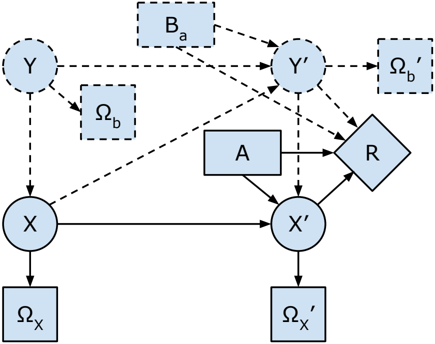
- represents emotions as vectors that represent evaluation (E), potency (P), and activity (A) respectively
- describes social events by an Actor-Behaviour-Object (ABO) grammar
- "fundamentals" of identities and behaviours; shared between people within a same culture
- "transient impressions": emotional feelings of people evoked by a specific event

The ACT Principal

Actors work to experience transient impressions that are consistent with their fundamental sentiments.

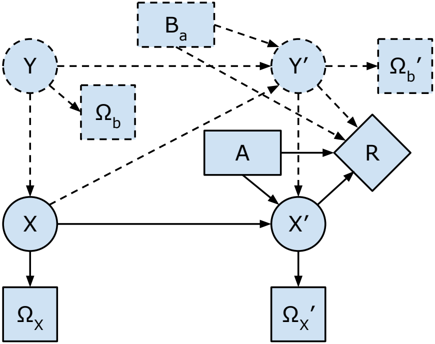
Concepts - POMDP

Partially Observable Markov Decision Process (POMDP)



- A timeslice of a POMDP process (solid lines)

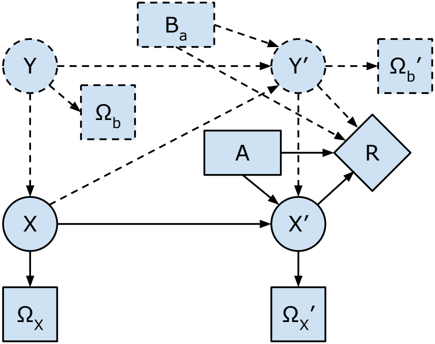
Partially Observable Markov Decision Process (POMDP)



- A timeslice of a POMDP process (solid lines)
- Variables: $\{\ X,\ A,\ \mathbf{\Omega_X}\ \}$

Concepts - POMDP

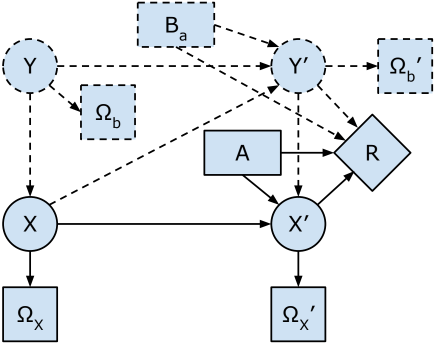
Partially Observable Markov Decision Process (POMDP)



- A timeslice of a POMDP process (solid lines)
- Variables: $\{ X, A, \mathbf{\Omega_X} \}$
- $Pr : X \to \Delta(\mathbf{\Omega_X})$,
  $Pr : X \times A \to \Delta(\mathbf{X})$

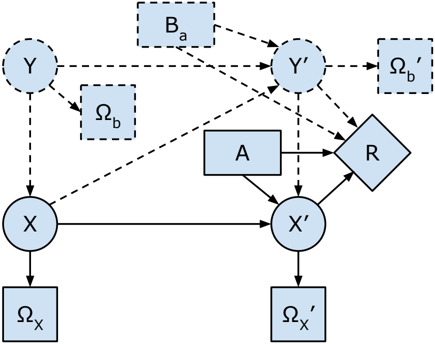
Concepts - POMDP

Partially Observable Markov Decision Process (POMDP)



- A timeslice of a POMDP process (solid lines)
- Variables: $\{ X, A, \mathbf{\Omega_X} \}$
- $Pr : X \to \Delta(\mathbf{\Omega_X})$,
  $Pr : X \times A \to \Delta(\mathbf{X})$
- Reward Function: $R(A, X')$

Partially Observable Markov Decision Process (POMDP)

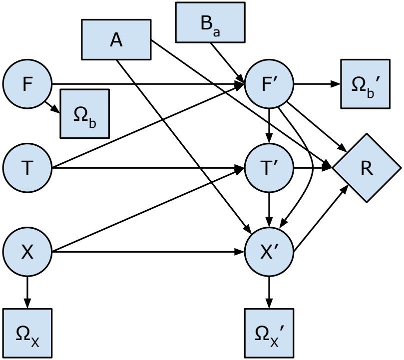


- A timeslice of a POMDP process (solid lines)
- Variables: $\{ X, A, \mathbf{\Omega_X} \}$
- $Pr : X \to \Delta(\mathbf{\Omega_X})$,
  $Pr : X \times A \to \Delta(\mathbf{X})$
- Reward Function: $R(A, X')$
- Augmented with affective states (dotted lines)

Concepts - BayesACT

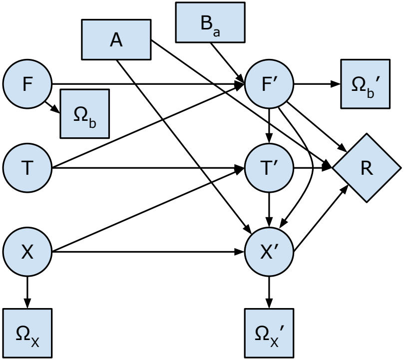
- A Bayesian version of the ACT theory
- Combines the ACT with POMDP model so that can learn an interactant's identity

- A Bayesian version of the ACT theory
- Combines the ACT with POMDP model so that can learn an interactant's identity



Concepts - BayesACT

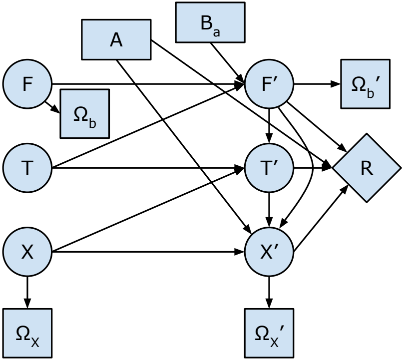
- A Bayesian version of the ACT theory
- Combines the ACT with POMDP model so that can learn an interactant's identity



- States $S = \{F, T, X\}$, where $F = \{F_{ij}\}$, $T = \{T_{ij}\}$, $i \in \{a, b, c\}$, $j \in \{e, p, a\}$

Concepts - BayesACT

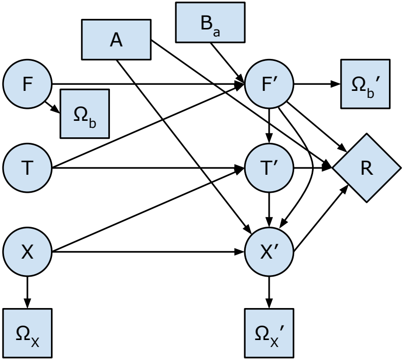
- A Bayesian version of the ACT theory
- Combines the ACT with POMDP model so that can learn an interactant's identity



- States $S = \{F, T, X\}$, where $F = \{F_{ij}\}$, $T = \{T_{ij}\}, i \in \{a, b, c\}, j \in \{e, p, a\}$
- Observations $\Omega = \{\Omega_X, \Omega_b\}$

Concepts - BayesACT

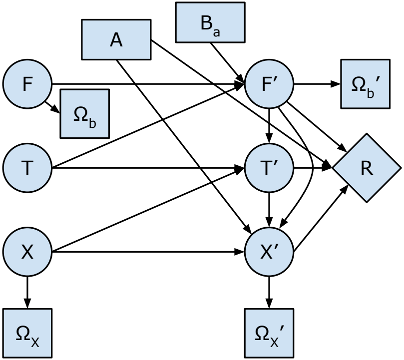
- A Bayesian version of the ACT theory
- Combines the ACT with POMDP model so that can learn an interactant's identity



- States $S = \{F, T, X\}$, where $F = \{F_{ij}\}, T = \{T_{ij}\}, i \in \{a, b, c\}, j \in \{e, p, a\}$
- Observations $\Omega = \{\Omega_X, \Omega_b\}$
- Actions $\{A, B_a\}$

Concepts - BayesACT

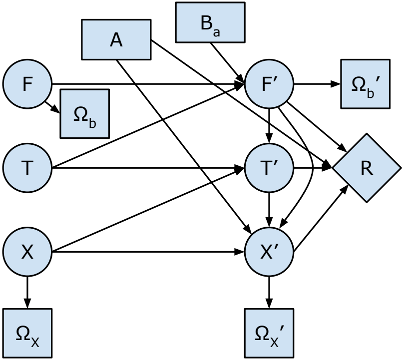
- A Bayesian version of the ACT theory
- Combines the ACT with POMDP model so that can learn an interactant's identity



- States $S = \{F, T, X\}$, where $F = \{F_{ij}\}$, $T = \{T_{ij}\}, i \in \{a, b, c\}, j \in \{e, p, a\}$
- Observations $\Omega = \{\Omega_X, \Omega_b\}$
- Actions $\{A, B_a\}$
- By updating F, the probability distribution of the client's identity F_c is learned

Concepts - BayesACT

- A Bayesian version of the ACT theory
- Combines the ACT with POMDP model so that can learn an interactant's identity



- States $S = \{F, T, X\}$, where $F = \{F_{ij}\}$, $T = \{T_{ij}\}$, $i \in \{a, b, c\}$, $j \in \{e, p, a\}$
- Observations $\Omega = \{\Omega_X, \Omega_b\}$
- Actions $\{A, B_a\}$
- By updating F, the probability distribution of the client's identity F_c is learned
- Calculate $\{A, B_a\}$ basing on $\{F, T, X\}$

Concepts - BayesACT cont.

Updates F and Calculates $\{A, B_a\}$ basing on $\{F, T, X\}$

Concepts - BayesACT cont.

Updates F and Calculates $\{A, B_a\}$ basing on $\{F, T, X\}$

- The deflection $\phi(F, T)$ between F and T:

$$\phi(f, t) \propto e^{-(f'-t')\Sigma^{-1}(f-t)} \tag{1}$$

Updates F and Calculates $\{A, B_a\}$ basing on $\{F, T, X\}$

- The deflection $\phi(F, T)$ between F and T:

$$\phi(f, t) \propto e^{-(f'-t')\Sigma^{-1}(f-t)} \tag{1}$$

- The probability of a post-action fundamental sentiment f':

$$Pr(f'|f, t, x, b_a, \phi) \propto e^{-\phi(f',t')-\xi(f',f,b_a,x)} \tag{2}$$

  where t' can be computed from $\{f', t, x\}$ by empirically derived prediction equations of ACT.

Concepts - BayesACT cont.

Updates F and Calculates $\{A, B_a\}$ basing on $\{F, T, X\}$

- The deflection $\phi(F, T)$ between F and T:

$$\phi(f, t) \propto e^{-(f'-t')\Sigma^{-1}(f-t)} \tag{1}$$

- The probability of a post-action fundamental sentiment f':

$$Pr(f'|f, t, x, b_a, \phi) \propto e^{-\phi(f',t')-\xi(f',f,b_a,x)} \tag{2}$$

  where t' can be computed from $\{f', t, x\}$ by empirically derived prediction equations of ACT.

- $Pr(x'|x, f', t', a)$: how the application progresses

Updates F and Calculates $\{A, B_a\}$ basing on $\{F, T, X\}$

- The deflection $\phi(F, T)$ between F and T:

$$\phi(f, t) \propto e^{-(f'-t')\Sigma^{-1}(f-t)} \tag{1}$$

- The probability of a post-action fundamental sentiment f':

$$Pr(f'|f, t, x, b_a, \phi) \propto e^{-\phi(f',t')-\xi(f',f,b_a,x)} \tag{2}$$

  where t' can be computed from $\{f', t, x\}$ by empirically derived prediction equations of ACT.

- $Pr(x'|x, f', t', a)$: how the application progresses
- $Pr(\omega_b|f)$ and $Pr(\omega_x|x)$: observation functions for the client behaviour sentiment and system state

Solution - Overview

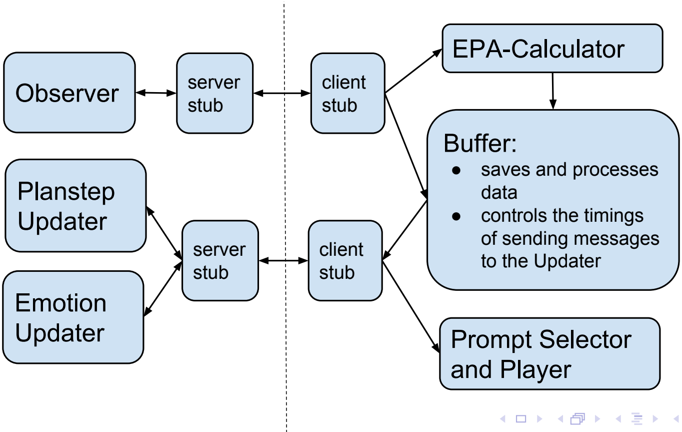
Goal

Design an *extensible* system that assists *people with dementia* during a hand-washing process by *assessing their states* and *provide instructions accordingly*.

Solution - Overview

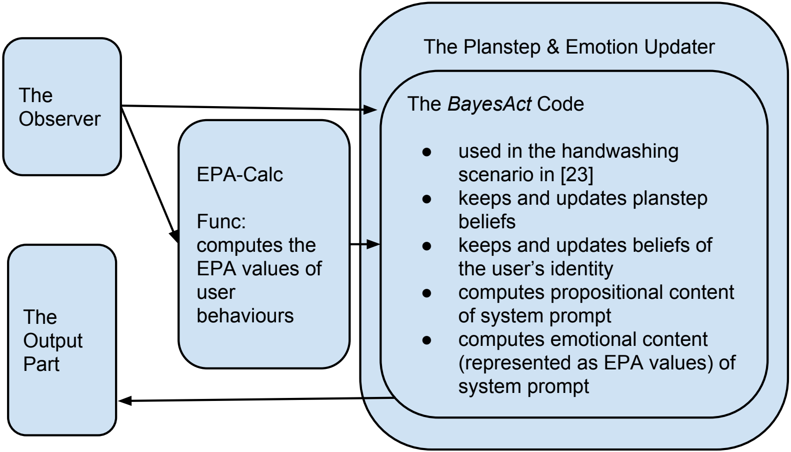
Goal

Design an *extensible* system that assists *people with dementia* during a hand-washing process by *assessing their states* and *provide instructions accordingly*.



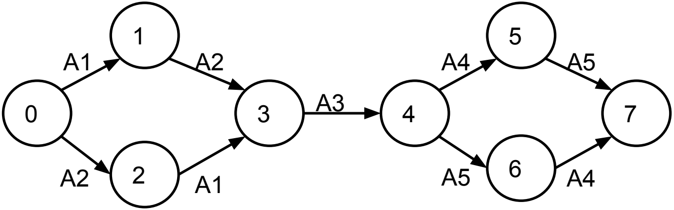
Solution - the Planstep and Emotion Updater

Design the Planstep and Emotion Updaters basing on the BayesAct code



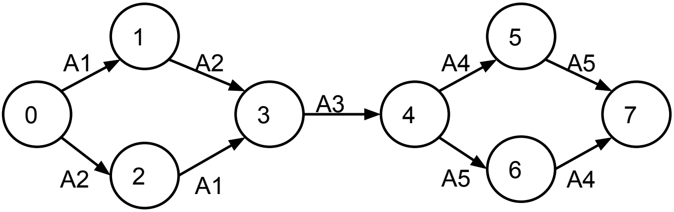
Solution - the Planstep and Emotion Updater cont.

A planstep update diagram



Solution - the Planstep and Emotion Updater cont.

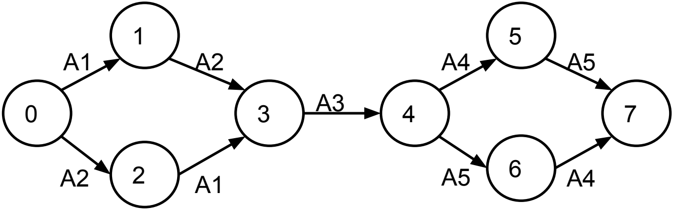
A planstep update diagram



- Eight plansteps: (0) "off/dirty/dry", (1) "on/dirty/dry", (2) "off/soapy/dry", (3) "on/soapy/dry", (4) "on/clean/wet", (5) "off/clean/wet", (6) "on/clean/dry", (7) "off/clean/dry"

Solution - the Planstep and Emotion Updater cont.

A planstep update diagram



- Eight plansteps: (0) “off/dirty/dry”, (1) “on/dirty/dry”, (2) “off/soapy/dry”, (3) “on/soapy/dry”, (4) “on/clean/wet”, (5) “off/clean/wet”, (6) “on/clean/dry”, (7) “off/clean/dry”
- Five behaviours: A1 to A5 are “turn on water”, “put on soap”, “rinse hands”, “turn off water”, and “use towel”, respectively.

Solution - the Planstep and Emotion Updater cont.

Use the BayesACT framework in the handwashing scenario

- Recall: BayesACT includes states $S = \{X, F, T\}$, observations $\Omega = \{\Omega_x, \Omega_b\}$, and agent actions $\{A, B_a\}$

Solution - the Planstep and Emotion Updater cont.

Use the BayesACT framework in the handwashing scenario

- Recall: BayesACT includes states $S = \{X, F, T\}$, observations $\Omega = \{\Omega_x, \Omega_b\}$, and agent actions $\{A, B_a\}$
- In our hand-washing system, $X = \{X_{turn}, X_{ps}, X_{aw}, X_{bahav}\}$

Solution - the Planstep and Emotion Updater cont.

Use the BayesACT framework in the handwashing scenario

- Recall: BayesACT includes states $S = \{X, F, T\}$, observations $\Omega = \{\Omega_x, \Omega_b\}$, and agent actions $\{A, B_a\}$
- In our hand-washing system, $X = \{X_{turn}, X_{ps}, X_{aw}, X_{bahav}\}$
- Ω_x gives evidence to the system about X

Solution - the Planstep and Emotion Updater cont.

Use the BayesACT framework in the handwashing scenario

- Recall: BayesACT includes states $S = \{X, F, T\}$, observations $\Omega = \{\Omega_x, \Omega_b\}$, and agent actions $\{A, B_a\}$
- In our hand-washing system, $X = \{X_{turn}, X_{ps}, X_{aw}, X_{bahav}\}$
- Ω_x gives evidence to the system about X
- Ω_b gives evidence to the system about f_b. The observation function $Pr(\Omega_b|f_b)$ allows one to specify the "confidence" or "reliability" of the different components of Ω_b by γ, which is the variance of a normal (Gaussian) distribution.

Solution - the Planstep and Emotion Updater cont.

Use the BayesACT framework in the handwashing scenario

- Recall: BayesACT includes states $S = \{X, F, T\}$, observations $\Omega = \{\Omega_x, \Omega_b\}$, and agent actions $\{A, B_a\}$
- In our hand-washing system, $X = \{X_{turn}, X_{ps}, X_{aw}, X_{bahav}\}$
- Ω_x gives evidence to the system about X
- Ω_b gives evidence to the system about f_b. The observation function $Pr(\Omega_b|f_b)$ allows one to specify the "confidence" or "reliability" of the different components of Ω_b by γ, which is the variance of a normal (Gaussian) distribution.
- Compute X'_{ps} based on observation Ω_x and $\{X_{ps}, X_{behav}, X_{aw}, F, T\}$

Use the BayesACT framework in the handwashing scenario

- Recall: BayesACT includes states $S = \{X, F, T\}$, observations $\Omega = \{\Omega_x, \Omega_b\}$, and agent actions $\{A, B_a\}$
- In our hand-washing system, $X = \{X_{turn}, X_{ps}, X_{aw}, X_{bahav}\}$
- Ω_x gives evidence to the system about X
- Ω_b gives evidence to the system about f_b. The observation function $Pr(\Omega_b|f_b)$ allows one to specify the “confidence” or “reliability” of the different components of Ω_b by γ, which is the variance of a normal (Gaussian) distribution.
- Compute X'_{ps} based on observation Ω_x and $\{X_{ps}, X_{behav}, X_{aw}, F, T\}$
- A denotes the propositional content of a system message; B_a denotes how the message should be expressed

Solution - the Planstep and Emotion Updater cont.

Use the BayesACT framework in the handwashing scenario

- Recall: BayesACT includes states $S = \{X, F, T\}$, observations $\Omega = \{\Omega_x, \Omega_b\}$, and agent actions $\{A, B_a\}$

Solution - the Planstep and Emotion Updater cont.

Use the BayesACT framework in the handwashing scenario

- Recall: BayesACT includes states $S = \{X, F, T\}$, observations $\Omega = \{\Omega_x, \Omega_b\}$, and agent actions $\{A, B_a\}$
- In our hand-washing system, $X = \{X_{turn}, X_{ps}, X_{aw}, X_{bahav}\}$

Solution - the Planstep and Emotion Updater cont.

Use the BayesACT framework in the handwashing scenario

- Recall: BayesACT includes states $S = \{X, F, T\}$, observations $\Omega = \{\Omega_x, \Omega_b\}$, and agent actions $\{A, B_a\}$
- In our hand-washing system, $X = \{X_{turn}, X_{ps}, X_{aw}, X_{bahav}\}$
- Ω_x gives evidence to the system about X

Use the BayesACT framework in the handwashing scenario

- Recall: BayesACT includes states $S = \{X, F, T\}$, observations $\Omega = \{\Omega_x, \Omega_b\}$, and agent actions $\{A, B_a\}$
- In our hand-washing system, $X = \{X_{turn}, X_{ps}, X_{aw}, X_{bahav}\}$
- Ω_x gives evidence to the system about X
- Ω_b gives evidence to the system about f_b. The observation function $Pr(\Omega_b|f_b)$ allows one to specify the "confidence" or "reliability" of the different components of Ω_b by γ, which is the variance of a normal (Gaussian) distribution.

Solution - the Planstep and Emotion Updater cont.

Use the BayesACT framework in the handwashing scenario

- Recall: BayesACT includes states $S = \{X, F, T\}$, observations $\Omega = \{\Omega_x, \Omega_b\}$, and agent actions $\{A, B_a\}$
- In our hand-washing system, $X = \{X_{turn}, X_{ps}, X_{aw}, X_{bahav}\}$
- Ω_x gives evidence to the system about X
- Ω_b gives evidence to the system about f_b. The observation function $Pr(\Omega_b|f_b)$ allows one to specify the "confidence" or "reliability" of the different components of Ω_b by γ, which is the variance of a normal (Gaussian) distribution.
- Compute X'_{ps} based on observation Ω_x and $\{X_{ps}, X_{behav}, X_{aw}, F, T\}$

Solution - the Planstep and Emotion Updater cont.

Use the BayesACT framework in the handwashing scenario

- Recall: BayesACT includes states $S = \{X, F, T\}$, observations $\Omega = \{\Omega_x, \Omega_b\}$, and agent actions $\{A, B_a\}$
- In our hand-washing system, $X = \{X_{turn}, X_{ps}, X_{aw}, X_{bahav}\}$
- Ω_x gives evidence to the system about X
- Ω_b gives evidence to the system about f_b. The observation function $Pr(\Omega_b|f_b)$ allows one to specify the "confidence" or "reliability" of the different components of Ω_b by γ, which is the variance of a normal (Gaussian) distribution.
- Compute X'_{ps} based on observation Ω_x and $\{X_{ps}, X_{behav}, X_{aw}, F, T\}$
- A denotes the propositional content of a system message; B_a denotes how the message should be expressed

Solution - the EPA-Calculator

- Essentially an "Affect Recognition" problem
- Learns/ Calculates affective interpretations from user behaviours

Solution - the EPA-Calculator

- Essentially an "Affect Recognition" problem
- Learns/ Calculates affective interpretations from user behaviours
- Feature Selection
  - analysis on facial expressions and speeches
  - application scenario and user-group constraints

Solution - the EPA-Calculator

- Essentially an "Affect Recognition" problem
- Learns/ Calculates affective interpretations from user behaviours
- Feature Selection
  - analysis on facial expressions and speeches
  - application scenario and user-group constraints
- Calculate P, A using piecewise linear interpolation method
  - The average distance between the user's two hands in a set of n neighbouring frames is scaled to P
  - The average distance the user's hands move between n neighbouring frames is scaled to A

Solution - the EPA-Calculator

- Essentially an "Affect Recognition" problem
- Learns/ Calculates affective interpretations from user behaviours
- Feature Selection
  - analysis on facial expressions and speeches
  - application scenario and user-group constraints
- Calculate P, A using piecewise linear interpolation method
  - The average distance between the user's two hands in a set of n neighbouring frames is scaled to P
  - The average distance the user's hands move between n neighbouring frames is scaled to A
- Temporally smoothing in the Buffer

$$X = \sum_{k=i}^{j} \left(\frac{alpha}{alpha+1}\right)^{j-k} * \frac{1}{alpha+1} * X[k] \tag{3}$$

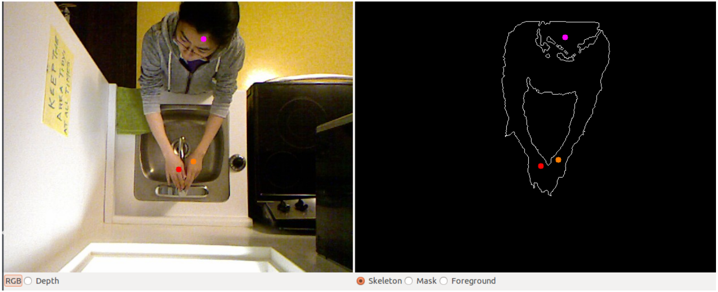
where $alpha \geq 0$, $X = P$ or $X = A$.

Solution - the Observer

- Step 1: Get the locations of the user's hands
  - utilize Czarnuch and Mihailidis's body tracker [**?**]
  - the tracker obtains body parts locations from the depth information of images taken from an overhead perspective
  - the tracker was trained using partially labeled, unbalanced data, and is configurable and re-trainable

Solution - the Observer

- Step 1: Get the locations of the user's hands
  - utilize Czarnuch and Mihailidis's body tracker [?]
  - the tracker obtains body parts locations from the depth information of images taken from an overhead perspective
  - the tracker was trained using partially labeled, unbalanced data, and is configurable and re-trainable



Solution - the Observer cont.

- Step 1: Get the locations of the user's hands
  - utilize Czarnuch and Mihailidis's body tracker [**?**]
  - the tracker obtains body parts locations from the depth information of images taken from an overhead perspective
  - the tracker was trained using partially labeled, unbalanced data, and is configurable and re-trainable
- Step 2: Map locations to user behaviours
  - compare hands locations with object positions
  - objects: the left-tap, the right-tap, the soap, the water-flow, the towel
  - observation noise handled by the observation function in the Reasoning Engine

Solution - the Observer cont.

- Step 1: Get the locations of the user's hands
  - utilize Czarnuch and Mihailidis's body tracker [?]
  - the tracker obtains body parts locations from the depth information of images taken from an overhead perspective
  - the tracker was trained using partially labeled, unbalanced data, and is configurable and re-trainable
- Step 2: Map locations to user behaviours
  - compare hands locations with object positions
  - objects: the left-tap, the right-tap, the soap, the water-flow, the towel
  - observation noise handled by the observation function in the Reasoning Engine
- Serves as an Observer server
  - with the help of the Buffer

Solution - the Output Part

- The prompt dataset: the audio-visual prompts generated and evaluated in Malhotra's study [?]
  - created 30 video clips using the USC Virtual Human Toolkit
  - EPA values of videos evaluated by human raters
  - the participants' answers are consistent with each other

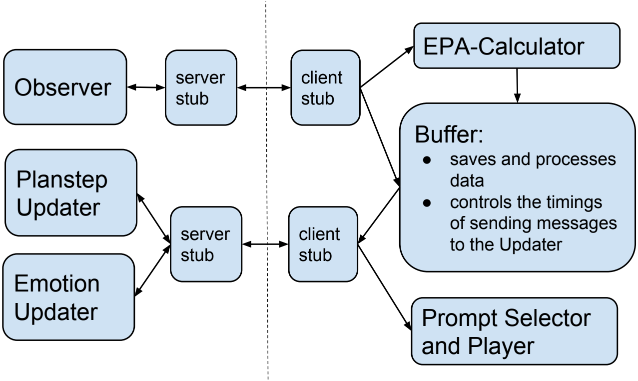

Solution - the Output Part

- The prompt dataset: the audio-visual prompts generated and evaluated in Malhotra's study [?]
  - created 30 video clips using the USC Virtual Human Toolkit
  - EPA values of videos evaluated by human raters
  - the participants' answers are consistent with each other

- A proper prompt is selected as the final prompt if it:
  - has the same propositional labels as the desired prompt
  - has the closest emotional (EPA) values as the desired prompt

Solution - Coordination between components

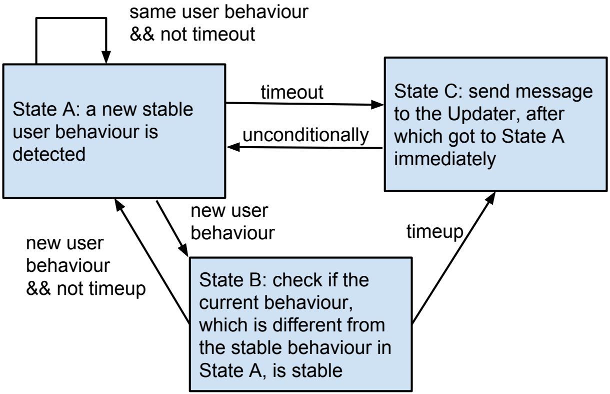
- The system is designed with independent components.



- How to coordinate between the components?
  - timings of sending request and response messages?

Solution - The Buffer

- Between the Observer, the EPA-Calc, and the Reasoning Engine
- Controls timings of sending messages



- Smoothes EPA values calucated by the Calculator

Experiments - Variables and Parameters

Experiments - Variables and Parameters cont.

Experiments - Test #1

Experiments - Test #1 cont.

Experiments - Test #2

Experiments - Test #2 cont.

Experiments - Conclusion

Discussion - Contribution

Discussion - Future Work

References

[1] The bayesact paper
[2] The tracker paper.
[3] The survey paper.

Acknowledgement

Jesse Hoey
James Tung and Peter van Beek
Xiao Yang, Chengbo Li and Enxun Wei

The end

Thank you!

- Questions?
- Comments?

Update X'_{ps} based on Ω_x and $\{X_{ps}, X_{behav}, X_{aw}, F, T\}$

- *SampleXVar*() and *evalSampleXVar*()
- Pseudocode of *SampleXVar*()

Update X'_{ps} based on Ω_x and $\{X_{ps}, X_{behav}, X_{aw}, F, T\}$ cont.

```
if Deflection(F, T) is high then
    threshold = high
else
    threshold = low
end if
if aw high then
    if prompted then
        if random_prob() < threshold then
            aw = low and not moving forward
        else if prompt wrong then
            aw = low and not moving forward
        else if likely then
            moving forward
        else if random_prob() < threshold then
            aw = low and not moving forward
        end if
    else
        if random_prob() < threshold then
            aw = low and not moving forward
        else
            aw stays high and moving forward
        end if
    end if
else
    if prompted then
        if random_prob() > threshold and prompt correct then
            move on and aw high
        else
            unlikely: aw high and not moving forward
        end if
    else
        unlikely: aw high and moving forward
    end if
end if
```

Update X'_{ps} based on Ω_x and $\{X_{ps}, X_{behav}, X_{aw}, F, T\}$ cont.

- $SampleXVar()$ and $evalSampleXVar()$
- Pseudocode of $SampleXVar()$

Update X'_{ps} based on Ω_x and $\{X_{ps}, X_{behav}, X_{aw}, F, T\}$ cont.

- $SampleXVar()$ and $evalSampleXVar()$
- Pseudocode of $SampleXVar()$
- $Pr : X_{behav} \rightarrow \Delta(\Omega_x)$ used in $evalSampleXVar()$