

SCC0284 / SCC5966

Sistemas de Recomendação

Aula 06: Filtragem Baseada em
Conhecimento

(mmanzato@icmc.usp.br)



Quando usar?

- Quando número de avaliações é baixo



Quando usar?

- Quando intervalo de tempo é importante
 - Itens que evoluem com rapidez

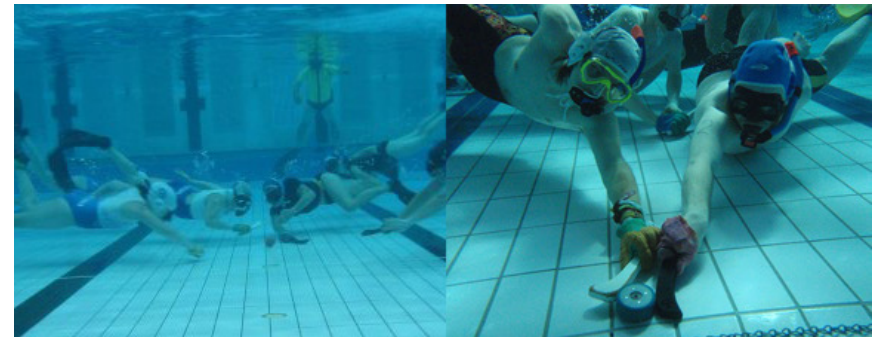


- Mudanças no estilo de vida ou situação familiar



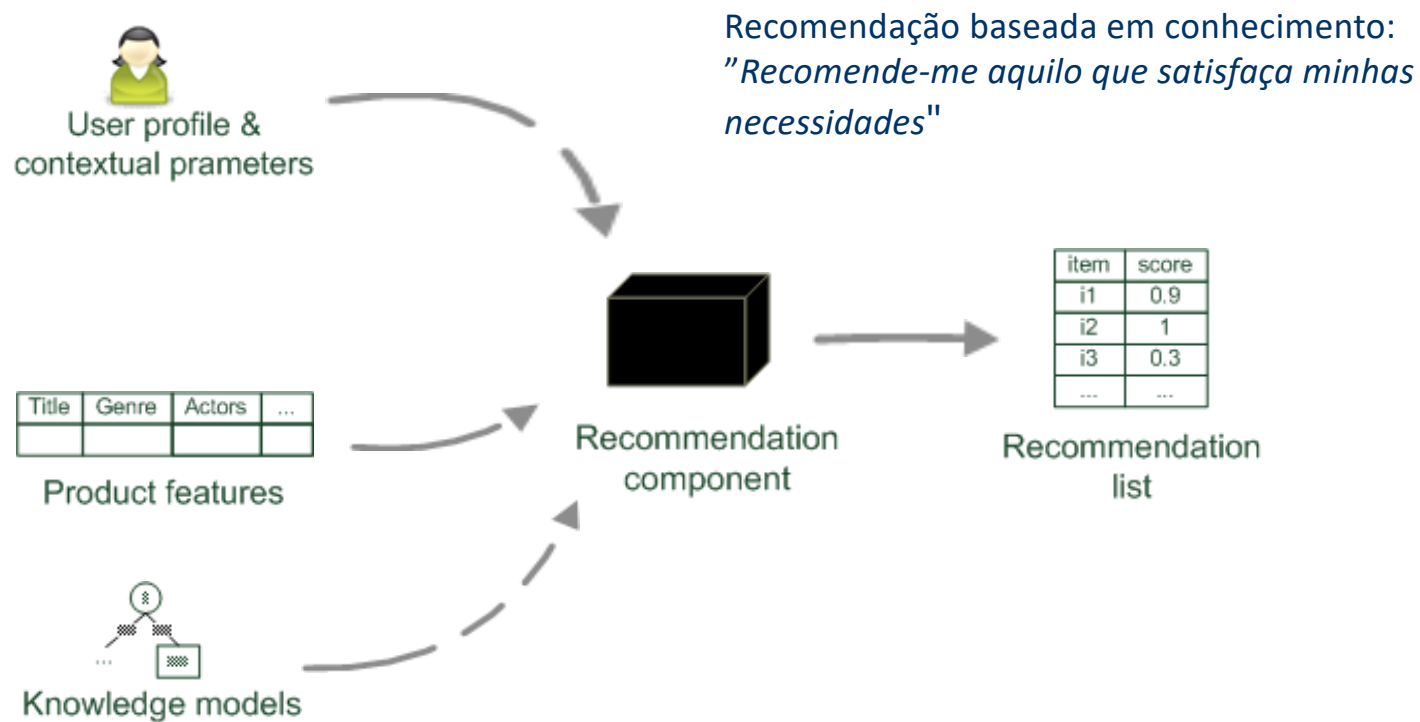
Quando usar?

- Quando usuários querem definir explicitamente suas necessidades
- Quando usuários não têm familiaridade com o domínio



Recomendação baseada em conhecimento

- Tipos de entrada em recomendação



Recomendação baseada em conhecimento

- Abordagens normalmente exploram o diálogo entre usuário e sistema
 - Usuários especificam os requisitos
 - Sistema identifica soluções, podendo explicá-las se solicitado
 - Se nenhuma solução satisfatória é encontrada, usuários alteram (parte dos) requisitos

Recomendação baseada em conhecimento

- Duas abordagens principais:
 - Baseada em **restrições**
 - Conjunto de regras é definido explicitamente
 - Recomendação deve seguir essas regras
 - Baseada em **casos**
 - Utiliza diferentes tipos de métricas de similaridade
 - Recuperação de itens similares aos requisitos formulados pelo usuário

Recomendação baseada em conhecimento

- Em geral, ambas abordagens têm conhecimento de domínio
 - Informações sobre requisitos do usuário e características dos itens

Domínio: câmeras digitais

id	price(€)	mpix	opt-zoom	LCD-size	movies	sound	waterproof
P ₁	148	8.0	4×	2.5	no	no	yes
P ₂	182	8.0	5×	2.7	yes	yes	no
P ₃	189	8.0	10×	2.5	yes	yes	no
P ₄	196	10.0	12×	2.7	yes	no	yes
P ₅	151	7.1	3×	3.0	yes	yes	no
P ₆	199	9.0	3×	3.0	yes	yes	no
P ₇	259	10.0	3×	3.0	yes	yes	no
P ₈	278	9.1	10×	3.0	yes	yes	yes

Ex. Requisitos do usuário:

✳ A câmera deve custar menos de 300 euros

✳ A câmera deve ser propícia para fotografar esportes

Baseada em restrições

- Recomendação baseada em restrições pode ser modelada como um CSP (*Constraint Satisfaction Problem*)
- Um CSP é definido como uma tripla (V, D, C), onde:
 - V : conjunto de variáveis
 - D : conjunto de domínios finitos para essas variáveis
 - C : conjunto de regras que descreve as combinações possíveis das variáveis
- Uma solução para CSP é a atribuição de um valor para cada variável em V que satisfaçam todas as regras em C

Exemplo

- Dada a tripla (V, D, C) , podemos expandir o modelo como:
 - $CSP (V = V_C \cup V_{PROD}, D, C = C_R \cup C_F \cup C_{PROD} \cup REQ)$
 - V_C : Variáveis do usuário
 - V_{PROD} : Variáveis do produto
 - C_R : Restrições do usuário
 - C_F : Compatibilidade entre restrições do usuário e características dos produtos
 - C_{PROD} : Características dos produtos
 - REQ : Requisitos do usuários

Exemplo

- CSP ($V = V_C \cup V_{PROD}$, D , $C = C_R \cup C_F \cup C_{PROD} \cup REQ$)
 - Dois conjuntos de variáveis (usuários e itens): $V = V_C \cup V_{PROD}$
 - $V_C : \{\text{min-opt-zoom}(3x..12x), \text{usage}(\text{digital}, \text{small-print}, \text{large-print}), \text{photography}(\text{sports}, \text{portrait}, \text{macro})\}$
 - $V_{PROD} : \{\text{price}(0..1000), \text{mpix}(3.0..12.0), \text{opt-zoom}(3x..12x), \text{lcd-size}(2.5..3.0), \text{movies}(\text{yes}, \text{no}), \text{sound}(\text{yes}, \text{no}), \text{waterproof}(\text{yes}, \text{no})\}$

id	price(€)	mpix	opt-zoom	LCD-size	movies	sound	waterproof
P ₁	148	8.0	4×	2.5	no	no	yes
P ₂	182	8.0	5×	2.7	yes	yes	no
P ₃	189	8.0	10×	2.5	yes	yes	no
P ₄	196	10.0	12×	2.7	yes	no	yes
P ₅	151	7.1	3×	3.0	yes	yes	no
P ₆	199	9.0	3×	3.0	yes	yes	no
P ₇	259	10.0	3×	3.0	yes	yes	no
P ₈	278	9.1	10×	3.0	yes	yes	yes

Exemplo

- CSP ($V = V_C \cup V_{PROD}$, D , $C = C_R \cup C_F \cup C_{PROD} \cup REQ$)
 - Três conjuntos de regras: $C = C_R \cup C_F \cup C_{PROD}$
 - $C_F : \{usage=large-print \rightarrow mpix > 7\}$ (compatibilidade usuário/item)
 - $C_R : \{photography=sports \rightarrow min-opt-zoom > 9x\}$ (restrições usuário)
 - $C_{PROD} : \{(id=p1 \wedge price=148 \wedge mpix=8.0 \wedge opt-zoom=4x \wedge lcd-size=2.5 \wedge movies=no \wedge sound=no \wedge waterproof=yes) \vee \dots \vee (id=p8 \wedge price=278 \wedge mpix=9.1 \wedge opt-zoom=10x \wedge lcd-size=3 \wedge movies=yes \wedge sound=yes \wedge waterproof=yes)\}$

id	price(€)	mpix	opt-zoom	LCD-size	movies	sound	waterproof
P ₁	148	8.0	4×	2.5	no	no	yes
P ₂	182	8.0	5×	2.7	yes	yes	no
P ₃	189	8.0	10×	2.5	yes	yes	no
P ₄	196	10.0	12×	2.7	yes	no	yes
P ₅	151	7.1	3×	3.0	yes	yes	no
P ₆	199	9.0	3×	3.0	yes	yes	no
P ₇	259	10.0	3×	3.0	yes	yes	no
P ₈	278	9.1	10×	3.0	yes	yes	yes

Exemplo

- CSP ($V = V_C \cup V_{PROD}$, D , $C = C_R \cup C_F \cup C_{PROD} \cup \text{REQ}$)
 - **REQ** : {usage=large-print, photography=sports}
 - usage=large-print \rightarrow mpix>7
 - photography=sports \rightarrow min-opt-zoom>9x
 - **RES** : {P₃, P₄, P₈}

id	price(€)	mpix	opt-zoom	LCD-size	movies	sound	waterproof
P ₁	148	8.0	4×	2.5	no	no	yes
P ₂	182	8.0	5×	2.7	yes	yes	no
P₃	189	8.0	10×	2.5	yes	yes	no
P₄	196	10.0	12×	2.7	yes	no	yes
P ₅	151	7.1	3×	3.0	yes	yes	no
P ₆	199	9.0	3×	3.0	yes	yes	no
P ₇	259	10.0	3×	3.0	yes	yes	no
P₈	278	9.1	10×	3.0	yes	yes	yes

Obs. Equivalente a realizar consultas conjuntivas numa tabela do banco de dados:

$$\delta_{[\text{mpix}>7, \text{opt-zom}>9x]}(P)$$

onde:

P : tabela do banco de dados

δ : operador de seleção

[mpix>7, opt-zoom>9x] : critérios de seleção

Baseada em casos

- Utilizam-se métricas de similaridade que indicam quanto as propriedades do item satisfazem os requisitos do usuário
- Dado um item p e um requisito $r \in \text{REQ}$:

$$\text{sim}(p, \text{REQ}) = \frac{\sum_{r \in \text{REQ}} w_r \cdot \text{sim}(p, r)}{\sum_{r \in \text{REQ}} w_r}$$

- onde:
 - w_r : peso de importância para o requisito r
 - $\text{sim}(p, r)$: é a distância entre o valor de atributo $\phi_r(p)$ (e.g. $\phi_{\text{mpix}}(p) = 8.0$) e o requisito r

Baseada em casos

- Na prática, há propriedades que o usuário deseja maximizar (e.g. mpixel) e propriedades que deseja minimizar (e.g. preço)
- Assim:

$$sim(p,r) = \frac{\phi_r(p) - \min(r)}{\max(r) - \min(r)}$$

P/ atributos que devem ser maximizados

$$sim(p,r) = \frac{\max(r) - \phi_r(p)}{\max(r) - \min(r)}$$

P/ atributos que devem ser minimizados

$$sim(p,r) = 1 - \frac{|\phi_r(p) - r|}{\max(r) - \min(r)}$$

P/ atributos que devem ser próximos dos requisitos do usuário

$$sim(p,r) = Jaccard(p,r)$$

P/ atributos multi-valorados (e.g. gêneros ou categorias de um item)

Baseada em casos

- Exemplo
 - Usando a filtragem baseada em casos, recomende uma lista de câmeras para o usuário segundo os seguintes requisitos:
 - $REQ = \{mpix > 8.0, price < 200, opt-zoom = 10x\}$
 - $w_{mpix} = w_{price} = w_{opt.zoom} = 1$

id	price(€)	mpix	opt-zoom	LCD-size	movies	sound	waterproof
P ₁	148	8.0	4×	2.5	no	no	yes
P ₂	182	8.0	5×	2.7	yes	yes	no
P ₃	189	8.0	10×	2.5	yes	yes	no
P ₄	196	10.0	12×	2.7	yes	no	yes
P ₅	151	7.1	3×	3.0	yes	yes	no
P ₆	199	9.0	3×	3.0	yes	yes	no
P ₇	259	10.0	3×	3.0	yes	yes	no
P ₈	278	9.1	10×	3.0	yes	yes	yes

Baseada em casos

- Exemplo
 - REQ = {mpix > 8.0, price < 200, opt-zoom = 10x}
 - Similaridade com p_1 :

id	price(€)	mpix	opt-zoom	LCD-size	movies	sound	waterproof
P ₁	148	8.0	4×	2.5	no	no	yes
P ₂	182	8.0	5×	2.7	yes	yes	no
P ₃	189	8.0	10×	2.5	yes	yes	no
P ₄	196	10.0	12×	2.7	yes	no	yes
P ₅	151	7.1	3×	3.0	yes	yes	no
P ₆	199	9.0	3×	3.0	yes	yes	no
P ₇	259	10.0	3×	3.0	yes	yes	no
P ₈	278	9.1	10×	3.0	yes	yes	yes

$$\begin{aligned} & \text{sim}(p_1, REQ) \\ &= \frac{\text{sim}(p_1, \text{mpix}) + \text{sim}(p_1, \text{price}) + \text{sim}(p_1, \text{opt.zoom})}{3} \\ &= \frac{\frac{8 - 7.1}{10 - 7.1} + \frac{278 - 148}{278 - 148} + 1 - \frac{|4 - 10|}{12 - 3}}{3} \\ &= 0.54 \end{aligned}$$

Interação e funcionamento

- Fluxo de interação:
 - 1) Usuário especifica suas preferências iniciais
 - 2) Quando requisitos suficientes forem coletados, o sistema calcula as recomendações
 - 3) Usuário pode revisar seus requisitos, ressubmetendo-os para cálculo de novas recomendações

Interação e funcionamento

- Fluxo de interação:
 - 1) Usuário especifica suas preferências iniciais
 - 2) Quando requisitos suficientes forem coletados, o sistema calcula as recomendações
 - 3) Usuário pode revisar seus requisitos, ressubmetendo-os para cálculo de novas recomendações
- E se o sistema não encontrar nenhuma resposta?

Interação e funcionamento

- Fluxo de interação:
 - 1) Usuário especifica suas preferências iniciais
 - 2) Quando requisitos suficientes forem coletados, o sistema calcula as recomendações
 - 3) Usuário pode revisar seus requisitos, ressubmetendo-os para cálculo de novas recomendações
- E se o usuário não souber especificar seus requisitos?

Interação e funcionamento

- Fluxo de interação:
 - 1) Usuário especifica suas preferências iniciais
 - 2) Quando requisitos suficientes forem coletados, o sistema calcula as recomendações
 - 3) Usuário pode revisar seus requisitos, ressubmetendo-os para cálculo de novas recomendações
- Como ranquear os itens selecionados de acordo com os requisitos?

Interação baseada em restrições

- Uso de valores padrão (*default values*)
 - Auxilia usuários que não sabem como especificar certos atributos
 - Podem ser usados para manipular usuários a escolherem certas opções
 - Tipos de valores padrão:
 - Estáticos
 - Dependentes
 - Derivados

Interação baseada em restrições

- Exemplo
 - Log de interação:

user	price	opt-zoom	lcd-size
u_1	400	10x	3.0
u_2	300	10x	3.0
u_3	150	4x	2.5
u_4	200	5x	2.7
u_5	200	5x	2.7

Únicos requisitos conhecidos para novo usuário: $\text{price} < 250 \wedge \text{opt-zoom} = 9x$

→ Que valor padrão poderia ser especificado para lcd-size?

Interação baseada em restrições

- Exemplo
 - Estratégia: 1-Nearest neighbor

$$\begin{aligned} & sim(u_1, REQ) \\ &= \frac{sim(u_1, price) + sim(u_1, opt.zoom)}{2} \\ &= \frac{\frac{\max(price) - \phi_{price}(u_1)}{\max(price) - \min(price)} + 1 - \frac{|\phi_{opt.zoom}(u_1) - opt.zoom|}{\max(opt.zoom) - \min(opt.zoom)}}{2} \\ &= \frac{\frac{400 - 400}{400 - 150} + 1 - \frac{|10 - 9|}{10 - 4}}{2} \\ &= 0.41 \end{aligned}$$

$$sim(u_2, REQ) = 0.61 \longrightarrow \text{lcd-size}=3.0$$

$$sim(u_3, REQ) = 0.58$$

$$sim(u_4, REQ) = 0.56$$

$$sim(u_5, REQ) = 0.56$$

Interação baseada em restrições

- Sugestão de requisitos
 - Nem todos usuários querem especificar valores para todas as propriedades
 - Porém, isso pode dificultar na escolha do melhor item
 - Pode-se usar o conceito de popularidade de acordo com requisições formuladas anteriormente (na sessão):

$$popularidade(atributo, pos) = \frac{\#seleções(atributo, pos)}{\#sessões}$$

onde:

- atributo : requisito especificado
- pos : ordem da interação na sessão

Interação baseada em restrições

- Exemplo

ID	pos: 1	pos: 2	pos: 3	pos: 4	pos: 5	pos: 6	...
1	price	opt-zoom	mpix	movies	lcd-size	sound	...
2	price	opt-zoom	mpix	movies	lcd-size	-	...
3	price	mpix	opt-zoom	lcd-size	movies	sound	...
4	mpix	price	opt-zoom	lcd-size	movies	-	...
5	mpix	price	lcd-size	opt-zoom	movies	sound	...

popularidade(price, pos: 1) = 0.6

popularidade(mpix, pos: 1) = 0.4

Assim, para o primeiro requisito, sugestiona-se o atributo *price*.

Interação baseada em restrições

- Problema de solução não encontrada
 - $\delta_{[\text{price} \leq 150, \text{opt-zoom} = 5x, \text{sound} = \text{yes}, \text{waterproof} = \text{yes}]}(P) = \emptyset$
 - Como resolver?

id	price(€)	mpix	opt-zoom	LCD-size	movies	sound	waterproof
P ₁	148	8.0	4×	2.5	no	no	yes
P ₂	182	8.0	5×	2.7	yes	yes	no
P ₃	189	8.0	10×	2.5	yes	yes	no
P ₄	196	10.0	12×	2.7	yes	no	yes
P ₅	151	7.1	3×	3.0	yes	yes	no
P ₆	199	9.0	3×	3.0	yes	yes	no
P ₇	259	10.0	3×	3.0	yes	yes	no
P ₈	278	9.1	10×	3.0	yes	yes	yes

Interação baseada em restrições

- Problema de solução não encontrada
 - Resolução por meio de diagnóstico baseado em modelo (MBD)
 - Dados $P = \{p_1, p_2, \dots, p_n\}$ e $REQ = \{r_1, r_2, \dots, r_m\}$, onde $\delta_{[REQ]}(P) = \emptyset$, calcula-se um conjunto de diagnósticos
$$\Delta = \{d_1, d_2, \dots, d_k\}$$
onde $\delta_{[REQ-d_i]}(P) \neq \emptyset$ para qualquer $d_i \in \Delta$.
 - Um diagnóstico é um conjunto mínimo de requisitos
$$\{r_1, r_2, \dots, r_l\} = d \subseteq REQ$$
que precisam ser alterados para que $\delta_{[REQ-d]}(P) \neq \emptyset$

Interação baseada em restrições

- Problema de solução não encontrada
 - O cálculo de um diagnóstico $d_i \in \Delta$ é baseado na determinação e resolução de conjuntos conflitantes (CS).
 - Um CS é um subconjunto mínimo $\{r_1, r_2, \dots, r_l\} \subseteq \text{REQ}$ tal que $\delta_{[\text{CS}]}(P) = \emptyset$.
 - CS mínimo é dado pela inexistência de um CS' tal que $\text{CS}' \subset \text{CS}$

Interação baseada em restrições

- Exemplo

id	price(€)	mpix	opt-zoom	LCD-size	movies	sound	waterproof
P ₁	148	8.0	4×	2.5	no	no	yes
P ₂	182	8.0	5×	2.7	yes	yes	no
P ₃	189	8.0	10×	2.5	yes	yes	no
P ₄	196	10.0	12×	2.7	yes	no	yes
P ₅	151	7.1	3×	3.0	yes	yes	no
P ₆	199	9.0	3×	3.0	yes	yes	no
P ₇	259	10.0	3×	3.0	yes	yes	no
P ₈	278	9.1	10×	3.0	yes	yes	yes

$$\delta_{[r1: \text{price} \leq 150, r2: \text{opt-zoom} = 5x, r3: \text{sound} = \text{yes}, r4: \text{waterproof} = \text{yes}]}(P) = \emptyset$$

Conjuntos conflitantes (mínimos):

CS1 = {r₁, r₂} (já que $\delta_{[CS1]}(P) = \emptyset$)

CS2 = {r₂, r₄} (já que $\delta_{[CS2]}(P) = \emptyset$)

CS3 = {r₁, r₃} (já que $\delta_{[CS3]}(P) = \emptyset$)

Interação baseada em restrições

- Exemplo

id	price(€)	mpix	opt-zoom	LCD-size	movies	sound	waterproof
P ₁	148	8.0	4×	2.5	no	no	yes
P ₂	182	8.0	5×	2.7	yes	yes	no
P ₃	189	8.0	10×	2.5	yes	yes	no
P ₄	196	10.0	12×	2.7	yes	no	yes
P ₅	151	7.1	3×	3.0	yes	yes	no
P ₆	199	9.0	3×	3.0	yes	yes	no
P ₇	259	10.0	3×	3.0	yes	yes	no
P ₈	278	9.1	10×	3.0	yes	yes	yes

$$\delta_{[r1: \text{price} \leq 150, r2: \text{opt-zoom} = 5x, r3: \text{sound} = \text{yes}, r4: \text{waterproof} = \text{yes}]}(P) = \emptyset$$

Conjuntos conflitantes (mínimos):

CS1 = {r₁, r₂} (já que $\delta_{[CS1]}(P) = \emptyset$)

CS2 = {r₂, r₄} (já que $\delta_{[CS2]}(P) = \emptyset$)

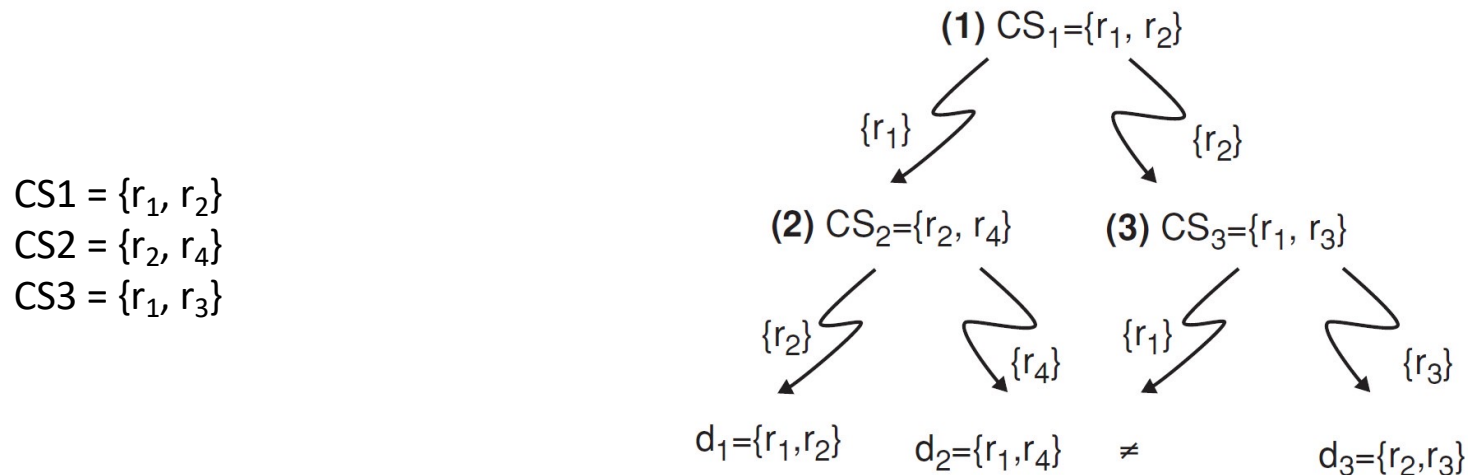
CS3 = {r₁, r₃} (já que $\delta_{[CS3]}(P) = \emptyset$)



Por ser mínimo, cada CS pode ser resolvido eliminando um de seus requisitos!

Interação baseada em restrições

- Exemplo



Assim, os diagnósticos derivados a partir dos conjuntos conflitantes (CS1, CS2 e CS3) são:
 $\Delta = \{d_1: \{r_1, r_2\}, d_2: \{r_1, r_4\}, d_3: \{r_2, r_3\}\}$

Interação baseada em restrições

- Para propor reparo a partir do conjunto de diagnósticos, basta fazer:
 - $\pi_{[\text{atributos}(d)]} \delta_{[\text{REQ-d}]}(P)$
- Exemplo: $d_1: \{r_1, r_2\}$

id	price(€)	mpix	opt-zoom	LCD-size	movies	sound	waterproof
P ₁	148	8.0	4×	2.5	no	no	yes
P ₂	182	8.0	5×	2.7	yes	yes	no
P ₃	189	8.0	10×	2.5	yes	yes	no
P ₄	196	10.0	12×	2.7	yes	no	yes
P ₅	151	7.1	3×	3.0	yes	yes	no
P ₆	199	9.0	3×	3.0	yes	yes	no
P ₇	259	10.0	3×	3.0	yes	yes	no
P ₈	278	9.1	10×	3.0	yes	yes	yes

$$\pi_{[\text{atributos}(d_1)]} \delta_{[\text{REQ-d}_1]}(P) =$$

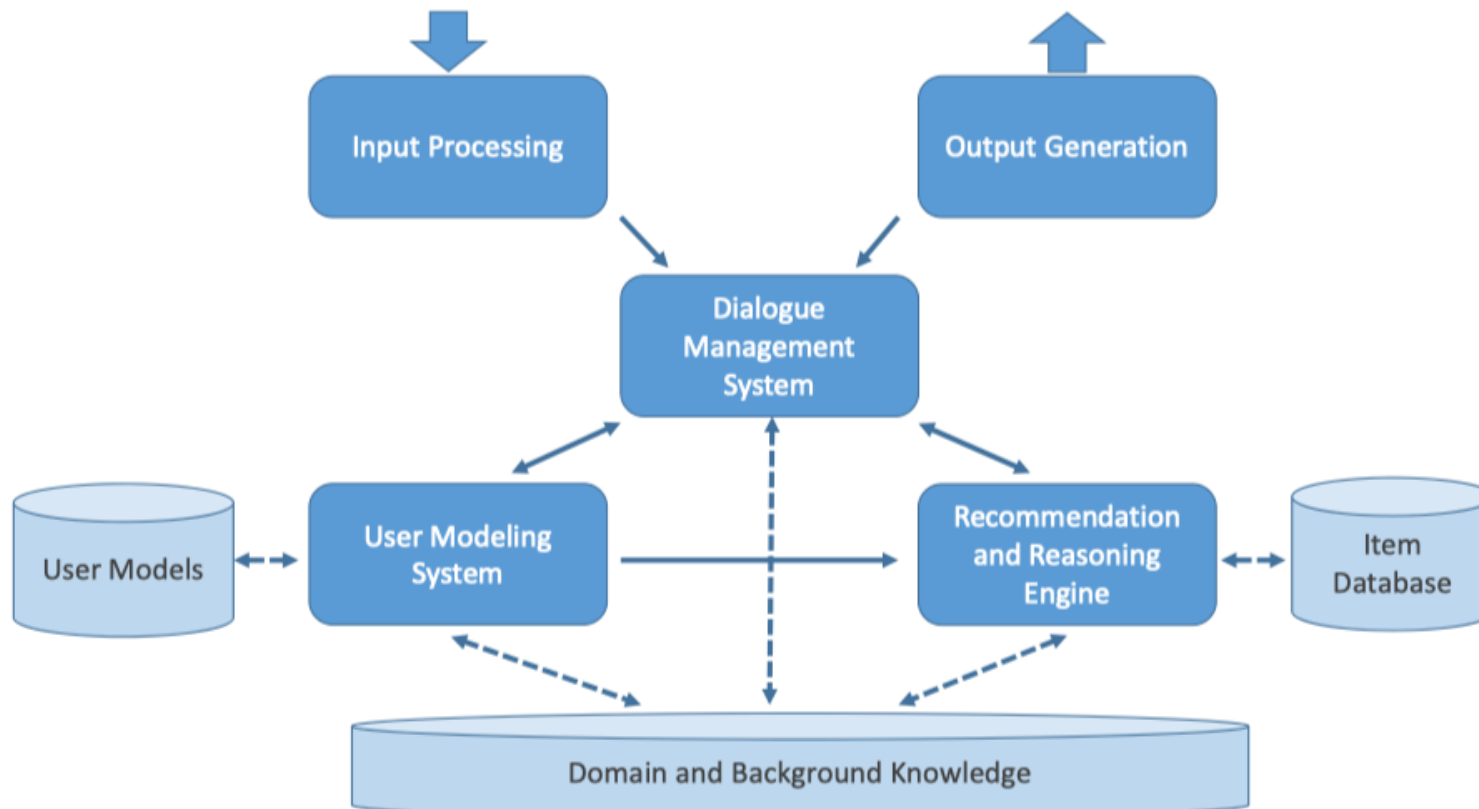
$$\pi_{[\text{price, opt-zoom}]} \delta_{[r_3: \text{sound=yes, } r_4: \text{waterproof=yes}]}(P) = \{\text{price}=278, \text{opt-zoom}=10x\}$$

Sistemas atuais...

- A filtragem baseada em conhecimento, atualmente, é mais conhecida como **Sistemas de Recomendação Conversacionais**
- Ganharam notoriedade devido ao avanço de assistentes pessoais (Siri, Alexa, etc.)
- Por outro lado:
 - Esses assistentes ainda carecem do estabelecimento de um diálogo com várias interações
 - Abordagens de SR conversacionais não se restringem a uma modalidade específica de interação



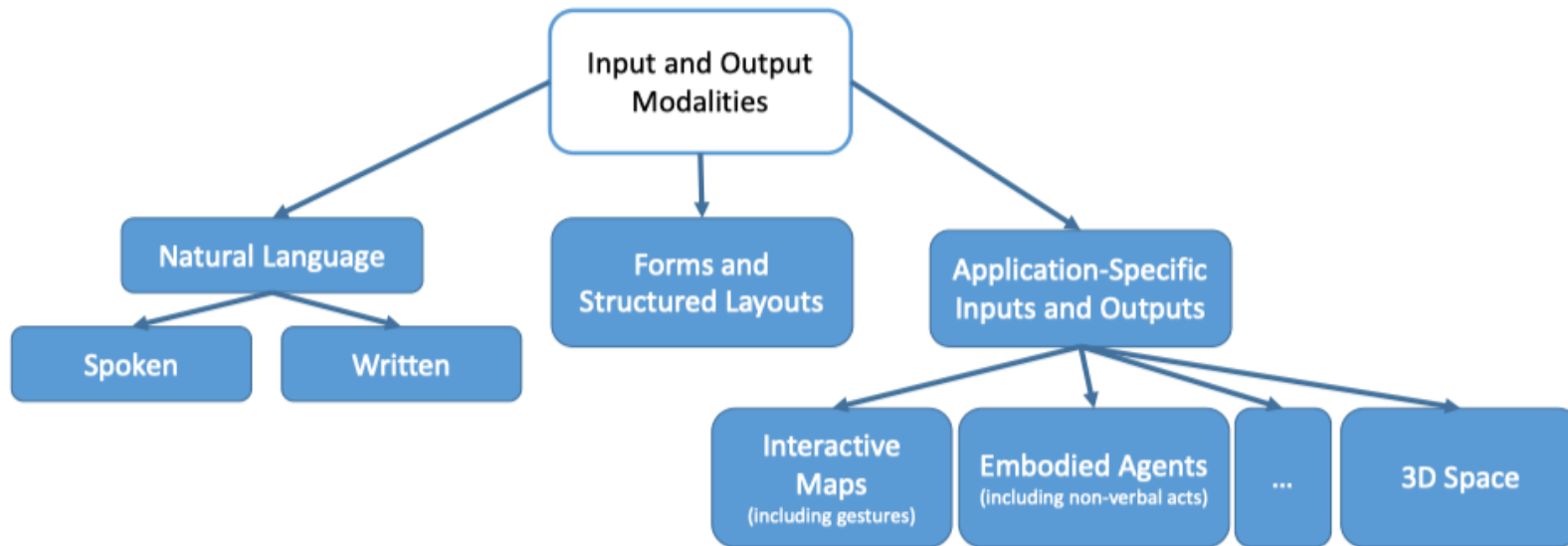
Arquitetura típica



Arquitetura típica

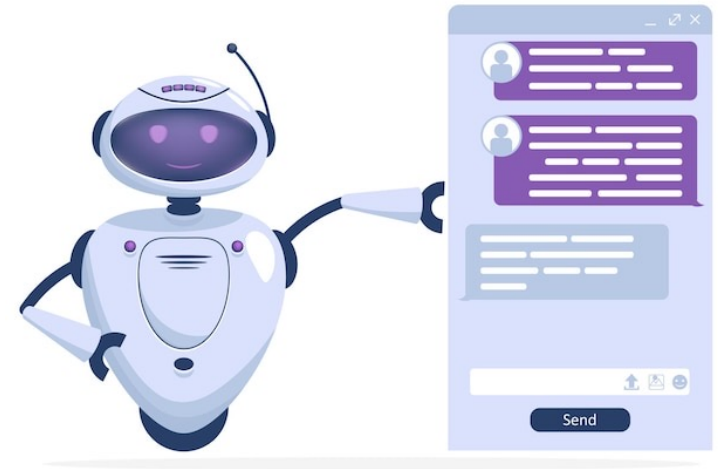
- Considerando a arquitetura típica, é possível explorar diferentes mecanismos implementados em cada módulo:
 - Entrada e saída (I/O)
 - Intenção do usuário
 - Modelagem das preferências
 - Estados possíveis do diálogo
 - Conhecimento de domínio
 - Uso de fontes externas (e.g. bases de conhecimento)

Modalidades de I/O



Intenção do usuário

- Durante o diálogo, o usuário pode:
 - Iniciar/finalizar diálogo
 - Chit-chat
 - Prover/revisar preferências
 - Pedir uma recomendação
 - Pedir uma explicação
 - Fornecer feedback da recomendação
 - Etc.

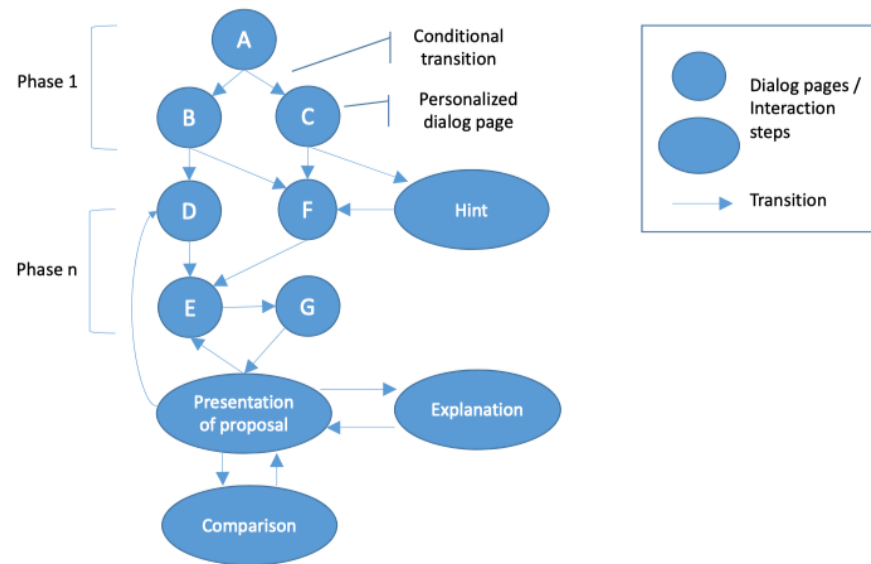


Modelagem das preferências

- Durante o diálogo, o sistema obtém as preferências (longo-curto prazo) do usuário
 - Preferências sobre um item
 - *“Vi o filme Curinga e achei o máximo!”*
 - Preferências sobre as facetas de um item
 - *“Não gosto muito de anime!”*

Estados possíveis do diálogo

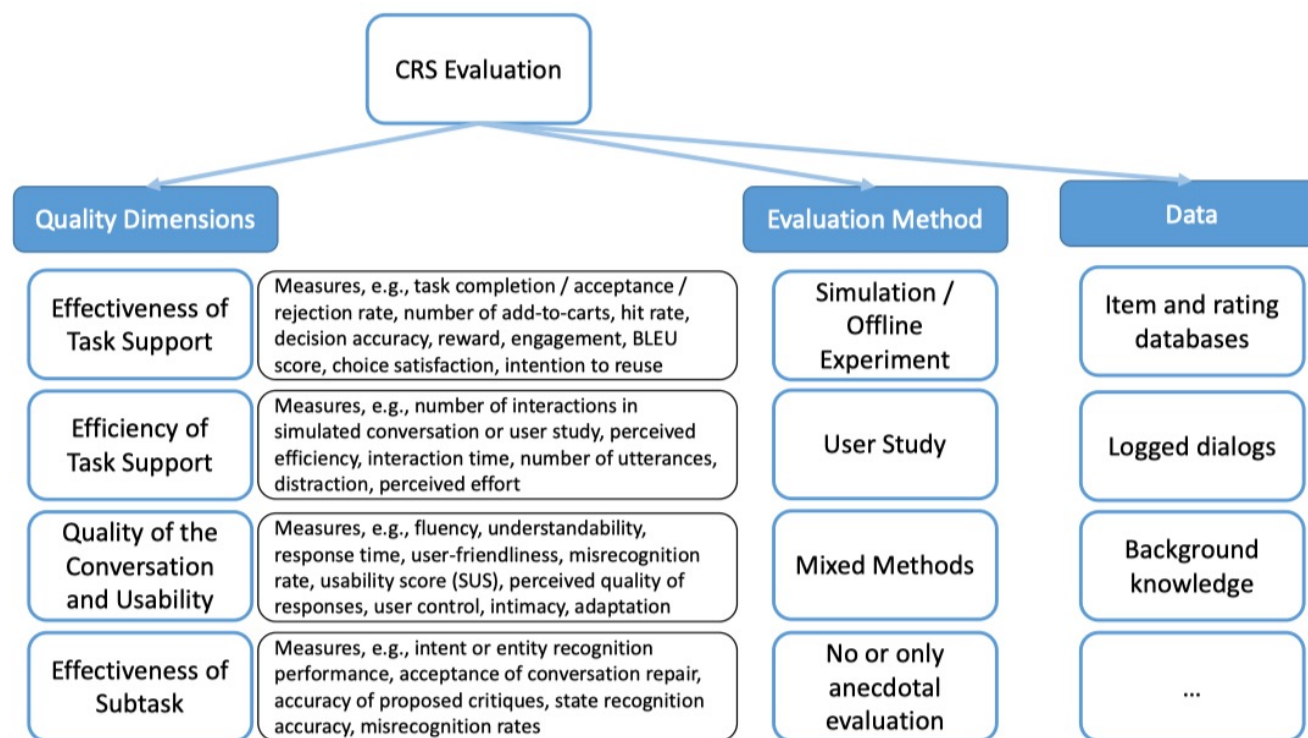
- O sistema precisa manter o controle do estado do diálogo para poder decidir a próxima ação
 - Geralmente, utilizam-se máquinas de estados finitas



Informações adicionais

- O sistema também pode utilizar:
 - Conhecimento de domínio
 - Bases de conversação
 - Histórico de interações
 - Bases de conhecimento (WikiData, WikiPedia, WordNet, etc.)
 - Etc.

Avaliação de SR Conversacionais



Recursos para desenvolvimento

- Chatbots
 - DialogFlow (Google)
 - Wit.ai (Facebook)
 - Watson Assistant (IBM)
 - Bot Framework (Microsoft)
- Bases de domínio
 - Movielens, Netflix, etc. (filmes)
 - Amazon reviews (vários domínios)
 - Last.fm, spotify, etc. (músicas)
 - Goodreads (livros)
- Bases de conversação
 - ReDial (filmes)
 - MultiWOZ (restaurantes/viagens)
 - MMD (fashion)
 - OpenDialKG (vários domínios)
- Bases de conhecimento
 - Wikipedia
 - WordNet
 - Wikiquote
 - Citysearch

Referências

- Dietmar Jannach, Markus Zanker, Alexander Felfernig, Gerhard Friedrich. *Recommender Systems: An Introduction*. Cambridge University Press, 2010.
- Dietmar Jannach, Ahtsham Manzoor, Wanling Cai, Li Chen. A Survey on Conversational Recommender Systems. Arxiv. 2020.

SCC0284 / SCC5966

Sistemas de Recomendação

Aula 06: Filtragem Baseada em
Conhecimento

(mmanzato@icmc.usp.br)

