

SCC0284 / SCC5966

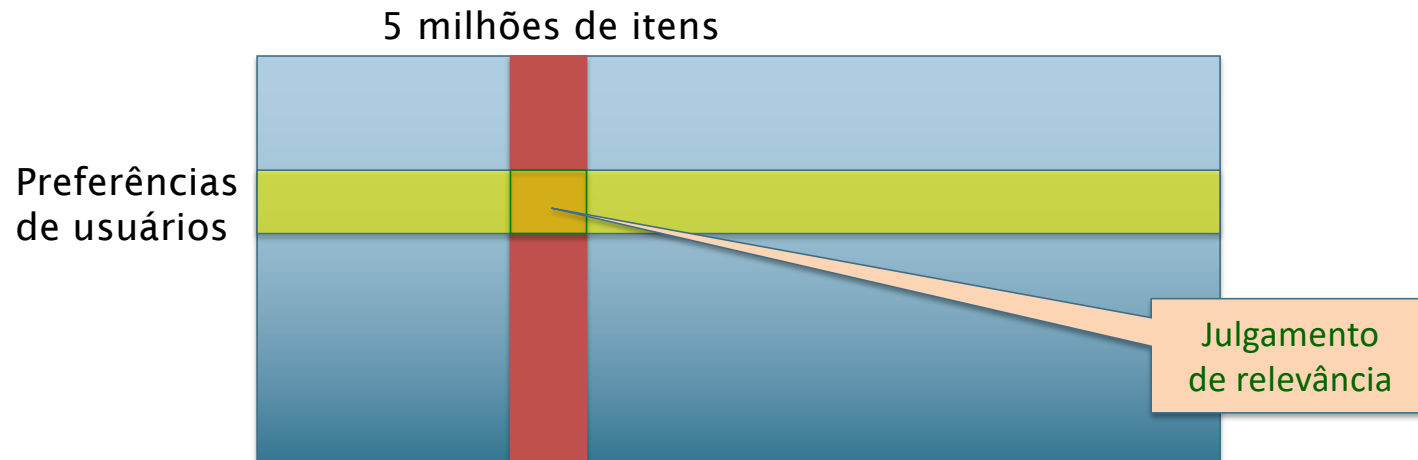
Sistemas de Recomendação

Aula 04: Avaliação em Sistemas de
Recomendação

(mmanzato@icmc.usp.br)

Introdução

- A avaliação de um sistema de recomendação implica em mensurar **o quão bem** o sistema atende às preferências do usuário
- Exemplo:



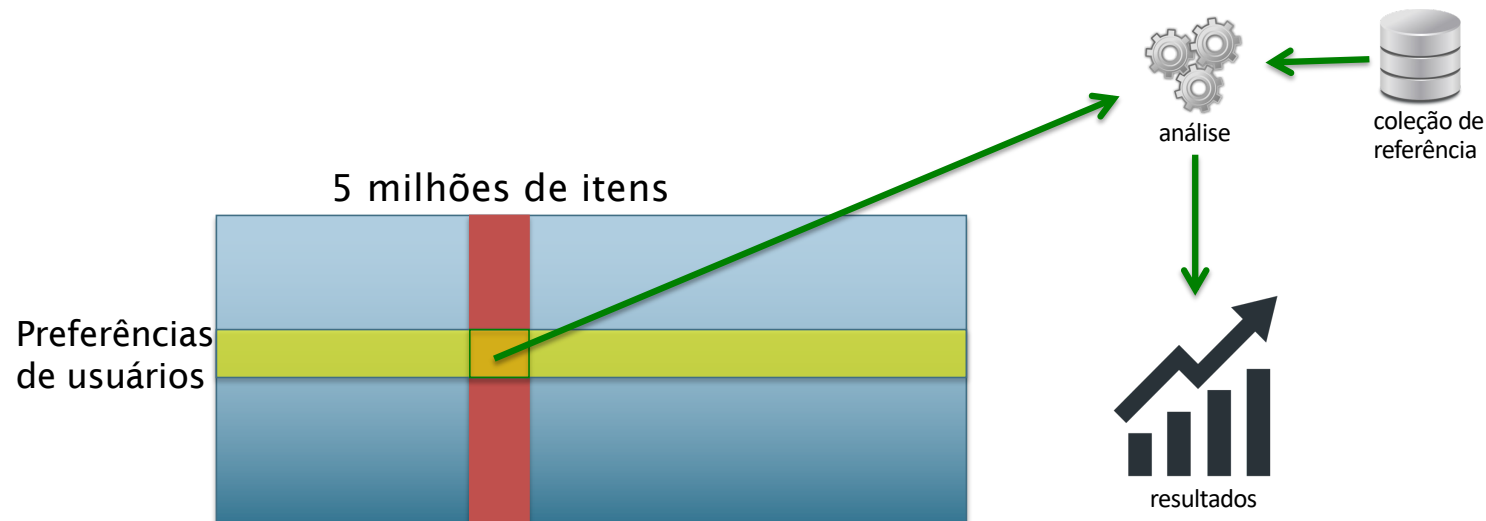
Introdução

- Sem uma avaliação sistemática de um sistema de recomendação, não é possível:
 - Determinar o quão bem um SR está desempenhando
 - Comparar o desempenho de um sistema com outros, de maneira objetiva



Introdução

- A avaliação da recomendação consiste em associar uma **métrica quantitativa** aos resultados produzidos por um SR



Introdução

- Avaliação pode ser feita usando três diferentes paradigmas:
 - Estudos com usuários
 - Avaliação online
 - Avaliação offline

Estudos com usuários

- Usuários são recrutados para interagir com o sistema sob avaliação
- Dados de interação e feedback são coletados e analisados
- Vantagem:
 - Permitem avaliar diferentes aspectos (acurácia, interface gráfica, explicações, etc.) sob a ótica do usuário
- Desvantagens:
 - Por estar sendo observado, o usuário pode enviesar sua opinião
 - Poucos usuários são usados, sem uma representação correta da população

Métodos online

- Avaliação a partir de recomendações realizadas para usuários reais
- Geralmente usuários não sabem que estão avaliando o sistema
- Limitações:
 - Necessidade de muitos usuários
 - Sistema como um todo não é publicamente acessível, limitando a avaliação apenas pela empresa/entidade responsável

Métodos online

- Exemplo: Teste A/B
 - Comparação de dois algoritmos utilizando a seguinte estratégia:
 1. Separar os usuários em dois grupos, A e B
 2. Cada grupo usa um algoritmo por um período de tempo
 3. Manter as demais configurações do sistema tão idênticas quanto possível
 4. No fim do processo, comparar a taxa de conversão de cada grupo

taxa de conversão: frequência com que um usuário seleciona um item que foi recomendado

Métodos offline

- Utilização de bases de dados contendo histórico de usuários
 - Inclusive bases de diversos domínios
- Avaliação a partir de interações já efetuadas
- Facilidade na reprodução dos experimentos
- Utilização de métricas que avaliam diferentes aspectos (acurácia, diversidade, novidade, cobertura, erro, etc.)

Métodos offline

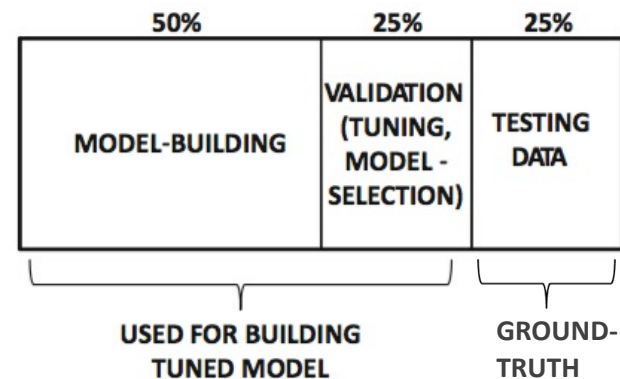
- Limitações:
 - Não consideram a propensão de usuários a reagirem às recomendações no futuro
 - Criação do conjunto-verdade (ground-truth) é feita a partir de suposições nem sempre verdadeiras
 - Não avalia outros aspectos do sistema, como interface, tempo de resposta, satisfação do usuário, etc.

Bases de Dados

- Devem conter (pelo menos):
 - Usuários, itens, interações (explícitas e/ou implícitas)
- Algumas fornecem dados adicionais:
 - Informações demográficas, metadados, tags, relações entre usuários, *timestamp*, etc.
- Algumas bases mais conhecidas na área:
 - <https://gab41.lab41.org/the-nine-must-have-datasets-for-investigating-recommender-systems-ce9421bf981c>
 - <https://gist.github.com/entaroadun/1653794>
 - <https://github.com/caserec/Datasets-for-Recommender-Systems>

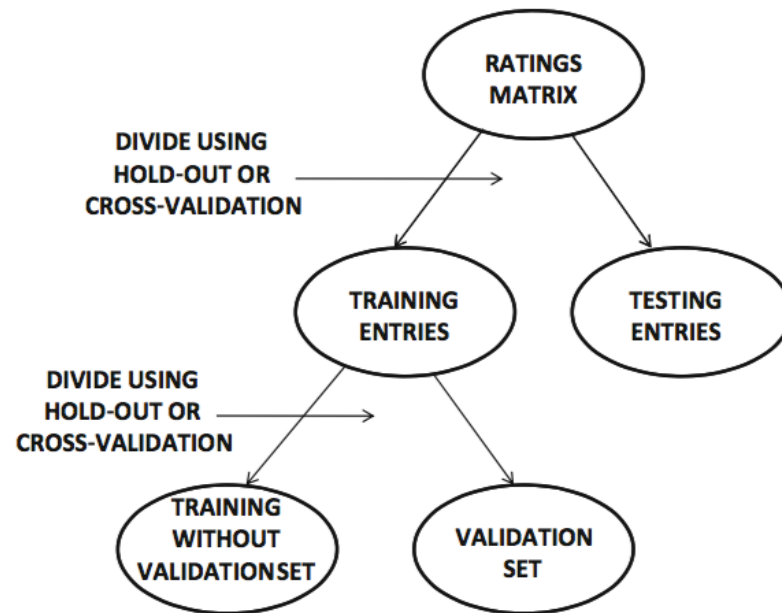
Projeto de avaliação offline

- Importante projetar o sistema de modo que os resultados não sejam enviesados
- Para isso, normalmente divide-se a base em três subconjuntos:
 - Conjunto de treinamento
 - Conjunto de validação
 - Conjunto de teste



Projeto de avaliação offline

- A divisão pode ser feita de duas maneiras:
 - Hold-Out
 - Cross-Validation



Projeto de avaliação offline

- Hold-Out
 - Uma fração da matriz de avaliações é separada para teste, e o restante usado para treinamento/validação
 - Avaliação é feita apenas com os dados de teste, os quais não podem ser usados para ajustes no modelo

Projeto de avaliação offline

- Cross-Validation
 - Matriz de avaliações é dividida em q conjuntos de igual tamanho
 - Seja S o conjunto de entradas na matriz R
 - O tamanho de cada conjunto será de $|S|/q$
 - Em uma rodada, um dos q conjuntos é usado para teste, e os $(q - 1)$ conjuntos restantes são usados para treinamento
 - São executadas q rodadas, cada uma alterando o conjunto de teste
 - No final o resultado da avaliação é a média de todas as rodadas

Projeto de avaliação offline

- (Exemplo) Cross-Validation



Projeto de avaliação offline

- Divisão de acordo com as avaliações dos usuários

All but N

- Separa N avaliações de cada usuário para o conjunto de teste
- Estabelece as mesmas condições para todos os usuários na fase de teste

Given N

- Mantém apenas N avaliações de cada usuário no conjunto de treinamento
- Mesma quantidade de informação para treinamento do perfil de cada usuário

Métricas de Acurácia

- A acurácia pode ser medida de duas formas:
 - Avaliação das **notas preditas** para cada par [usuário-item]
 - Análise do erro de predição
 - Avaliação de **rankings de itens** recomendados para os usuários
 - Apenas os top-N itens são avaliados
- Em ambos os casos, utiliza-se o conjunto de teste para medir

Predição de Avaliações

- Dados:
 - S : conjunto de avaliações observáveis
 - E : conjunto de avaliações usado para teste onde $E \subset S$
 - r_{ui} : avaliação real do par usuário-item (u, i)

Erro de predição:

$$\varepsilon_{ui} = r_{ui} - \hat{r}_{ui}$$

Root Mean Squared Error (RMSE):

$$RMSE = \sqrt{\frac{\sum_{(u,i) \in E} \varepsilon_{ui}^2}{|E|}}$$

Mean Absolute Error (MAE):

$$MAE = \frac{\sum_{(u,i) \in E} |\varepsilon_{ui}|}{|E|}$$

Predição de Avaliações

- Exemplo

Nr.	UserID	MovieID	Rating (r_i)	Prediction (p_i)	$ p_i - r_i $	$(p_i - r_i)^2$	
1	1	134	5	4.5	0.5	0.25	✗
2	1	238	4	5	1	1	✗
3	1	312	5	5	0	0	
4	2	134	3	5	2	4	✗
5	2	767	5	4.5	0.5	0.25	✗
6	3	68	4	4.1	0.1	0.01	
7	3	212	4	3.9	0.1	0.01	
8	3	238	3	3	0	0	
9	4	68	4	4.2	0.2	0.04	
10	4	112	5	4.8	0.2	0.04	
					4.6	5.6	

Toda a tabela:

- MAE = 0.46
- RMSE = 0.75

Removendo linha 4:

- MAE = 0.29
- RMSE = 0.42

Removendo linhas
1,2,4,5:

- MAE = 0.1
- RMSE = 0.13

Predição de Avaliações

- RMSE vs. MAE
 - Como RMSE soma erros ao quadrado, ele é mais afetado por outliers ou valores de erro elevados
 - Em termos de avaliação de robustez, RMSE é mais indicado
 - MAE, por outro lado, reflete melhor o erro médio
- Ambas as métricas, entretanto, não conseguem mensurar o impacto da melhoria da recomendação para o usuário

Ranqueamento

- Na avaliação de ranqueamento, podemos inicialmente criar uma matriz de confusão:

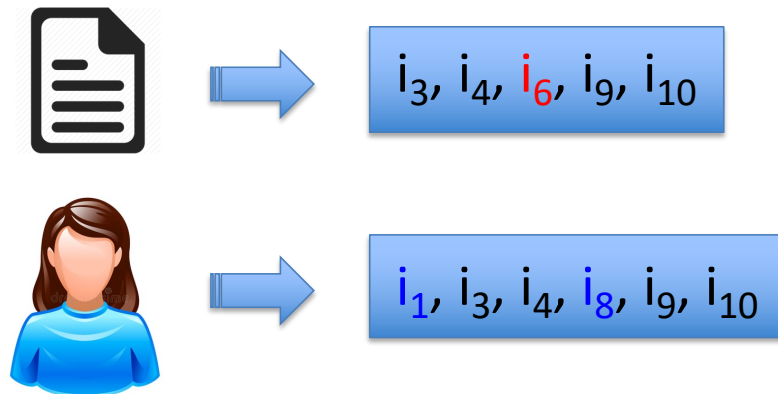
		Realidade	
		Relevante	Irrelevante
Recomen- dou?	Sim	True Positive (tp)	False Positive (fp)
	Não	False Negative (fn)	True Negative (tn)

}] **Todos os itens
recomendados (A)**

[**Todos os itens
relevantes (R)**

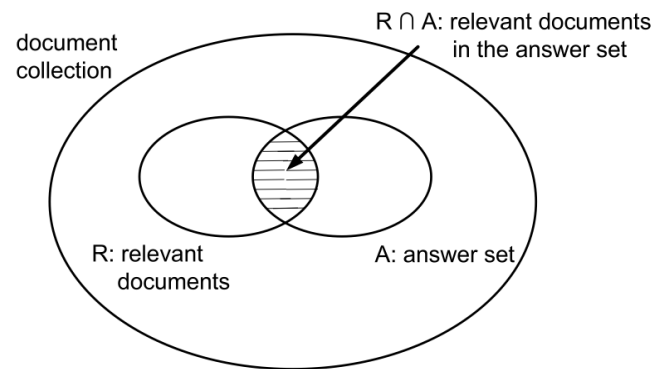
Ranqueamento

- Construção do conjunto-verdade
 - Utiliza conjunto de teste
 - Suposição: usuários tendem a avaliar/visitar apenas itens relevantes



Ranqueamento

- Considere:
 - **R** : conjunto de itens relevantes, $R = tp + fn$
 - **A** : o conjunto de itens recomendados, $A = tp + fp$
 - **$R \cap A$** : a intersecção dos conjuntos **R** e **A**



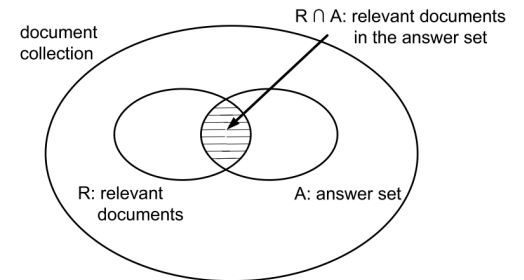
Precisão e Revocação

- As métricas revocação e precisão são definidas como:
 - **Revocação**: fração de itens relevantes que foram recomendados

$$Revocação = \frac{|R \cap A|}{|R|} = \frac{tp}{tp + fn}$$

- **Precisão**: fração de itens recomendados que são relevantes

$$Precisão = \frac{|R \cap A|}{|A|} = \frac{tp}{tp + fp}$$



Precisão e Revocação

- As métricas de revocação e precisão se complementam
- Exemplos
 - Sistema com **baixa** revocação e **alta** precisão
 - Sistema com **alta** revocação e **baixa** precisão
- Existem métricas que combinam ambas medidas em uma só
 - Exemplo: F1

$$F1 = 2 \frac{\text{precisão} * \text{revocação}}{\text{precisão} + \text{revocação}}$$

P@N

- Geralmente, o que mais importa para o usuário é a quantidade de itens relevantes **no topo** do ranking
- Precisão em **N** mede o desempenho quando **N** itens foram examinados
 - Geralmente, $N = \{1, 5, 10, 20\}$

P@5 e P@10

- Exemplo:


01. d_{123} •	06. d_9 •	11. d_{38}
02. d_{84}	07. d_{511}	12. d_{48}
03. d_{56} •	08. d_{129}	13. d_{250}
04. d_6	09. d_{187}	14. d_{113}
05. d_8	10. d_{25} •	15. d_3 •

- Nessa consulta, $P@5=40\%$ e $P@10=40\%$
- É possível computar a média $P@5$ e $P@10$ para M usuários
- Indicam qual algoritmo é preferível **aos olhos do usuário**


AP

- Average Precision ou Precisão Média (AP)
- Enfatiza itens relevantes que foram retornados no topo do ranking
- Na prática, é a média das precisões calculadas em cada item relevante retornado

Rank	Hit?
1	
2	X
3	X
4	X
5	


$$AP = \frac{1}{3} \left(\frac{1}{2} + \frac{2}{3} + \frac{3}{4} \right) = \frac{23}{36} \approx 0.639$$

$$AP = \frac{1}{3} \left(\frac{1}{1} + \frac{2}{4} + \frac{3}{5} \right) = \frac{21}{30} = 0.7$$



Rank	Hit?
1	X
2	
3	
4	X
5	X

MAP

- *Mean Average Precision* ou Média das Precisões Médias
- Média dos valores de AP para todos os rankings/usuários

$$MAP = \frac{1}{|U|} \sum_{u=1}^U AP_u$$

Ranqueamento

- Dilema com avaliação de rankings em SR

Avaliação offline	Avaliação online
Notas, transações	Notas, feedback
Histórico de interações (nem todos os itens recomendados são avaliados)	Interações ao vivo (todos os itens recomendados são avaliados)
Predições de itens não avaliados são desconhecidas, e são interpretados como "não relevantes" (suposição inicial: usuário tende a avaliar apenas itens relevantes)	Predições de itens relevantes e não relevantes que NÃO foram recomendados são desconhecidas
Se a suposição inicial não for verdadeira: tp pode ser muito baixo; fn pode ser muito baixo	fn e tn não podem ser determinados
Precisão e revocação podem variar	Precisão ok; Revocação questionável

Ranqueamento

- Avaliação offline
 - Efeito da suposição inicial



$i_3, i_4, i_6, i_9, i_{10}$

conjunto-verdade



$i_1, i_3, i_4, i_8, i_9, i_{10}$

realidade

Recomendação:

$i_1, i_4, i_9, i_{11}, i_{12}$

$$P = 2/(2+3) = 0.4$$

$$P = 3/(3+2) = 0.6$$

Ranqueamento

- Avaliação offline
 - Efeito da suposição inicial



$i_3, i_4, i_6, i_9, i_{10}$

conjunto-verdade



$i_1, i_3, i_4, i_8, i_9, i_{10}$

realidade

Recomendação:

$i_3, i_6, i_9, i_{10}, i_{11}$

$$P = 4/(4+1) = 0.8$$

$$P = 3/(3+2) = 0.6$$

Outros aspectos de avaliação

- A avaliação em SR pode ter como objetivo mensurar os seguintes aspectos:

Cobertura

Confiança

Novidade

Serendipidade

Diversidade

Robustez /
estabilidade

Escalabilidade

Cobertura

- Fração de usuários para os quais pelo menos **N** previsões podem ser calculadas
 - **N** : tamanho da lista de recomendações
- Fração de itens cujas avaliações de pelo menos **N** usuários podem ser previstas
- Necessário usar essa métrica em conjunto com acurácia
 - Recomendações poderiam ser geradas aleatoriamente
- **Pergunta**: qual estratégia simples que poderia ser usada para melhorar a cobertura?

Confiança

- Indica a convicção do sistema ou do usuário nas recomendações geradas
- Pode ser obtida por meio de explicações, relações de confiança, intervalo de confiança, etc.
- Pode-se melhorar a confiança em detrimento a outras medidas (e.g. novidade/surpresa)
- Maneira mais simples de medir confiança é por meio de experimentos online

Novidade

- Capacidade do sistema em recomendar itens desconhecidos, porém relevantes
- Recomendar itens desconhecidos aumenta a utilidade do sistema e melhora a taxa de conversão
- Métricas offline precisam da informação temporal das avaliações
- Possível medir também com avaliação online

Serendipidade

- Mede o nível de surpresa do usuário em **recomendações bem sucedidas**
- É mais rigorosa do que novidade: toda recomendação “serendipiosa” possui novidade, mas não vice-versa
 - Além de novo, o item recomendado **não é esperado**
- Exemplos:
 - Usuário gosta de comida japonesa, e o sistema recomenda um novo restaurante japonês
 - Há novidade, mas a recomendação é óbvia
 - Usuário gosta de comida japonesa, e o sistema recomenda comida chinesa
 - Há novidade, e a recomendação não é óbvia

Diversidade

- Mede o quanto os itens pertencentes a uma mesma lista de recomendações são diferentes entre si
- Se um usuário não gosta de um item recomendado, pode ser que ele goste de outro da mesma lista
- Aumentar a diversidade pode aumentar a novidade e a serendipidade
- Pode-se medir por meio de métricas de similaridade entre os itens

Robustez / estabilidade

- Mede o quanto o sistema é estável e robusto contra ataques ou avaliações falsas de usuários maliciosos
- Um sistema pode ser alvo de ataque se, por exemplo, o objetivo for aumentar o lucro de um produto com concorrência

Escalabilidade

- Qual o desempenho do sistema com bases extensas, em termos de eficiência e eficácia
- Métricas podem avaliar:
 - Tempo de treinamento
 - Tempo de predição / recomendação
 - Requisitos de memória
 - Etc.

Referências

- Dietmar Jannach, Markus Zanker, Alexander Felfernig, Gerhard Friedrich. *Recommender Systems: An Introduction*. Cambridge University Press, 2010.
- Aggarwal, Charu C. *Recommender Systems: The Textbook*. Springer, 2016.